A Semantic Approach to Evaluate the Impact of Cyber Actions on the Physical Domain

Alexandre de Barros Barreto
Instituto Tecnológico de Aeronáutica
São José dos Campos, SP, Brazil
Email: adebarro@c4i.gmu.edu

Paulo Cesar G. Costa
George Mason University
Fairfax, VA, USA
Email: pcosta@gmu.edu

Edgar T. Yano
Instituto Tecnológico de Aeronáutica
São José dos Campos, SP, Brazil
Email: yano@ita.br

Abstract—**Evaluating the impact that events within the cyber domain have on a military operation and its critical infrastructure is a non-trivial question, which remains unanswered so far in spite of the various research efforts addressing it. The key issue underlying this question is the difficulty in correlating cyber and physical behaviors in an integrated view, thus allowing for real-time analysis. This paper addresses the issue with the development of an ontology-based framework in which the cyber and physical behaviors are integrated in a consolidated view, using a combination of open standards protocols and semantic technologies. In our approach, the mission and its physical aspects are modeled using a business process language (e.g., BPMN) and an information infrastructure based on Simple Network Management Protocol (SNMP). In this scheme, changes in the environment are captured using the output of sensor components existing in the infrastructure. In order to ensure a complete and integrated analysis of the accruing data, we have developed a Cyber Situation ontology (in OWL) and a methodology for mapping the cyber and the physical domains. In this framework, mission data from the environment is retrieved and fused using an engine based on the Semantic Web Rule Language (SWRL). The output of this process is then presented to an analyst in a way that only the most important information needed to support his/her decisions is shown. To validate our approach, a real air traffic scenario was modeled and many simulated flights were generated to support of our experiments.**

I. INTRODUCTION

With the increasing automation of processes and systems that are part of critical infrastructures supporting military and vital civilian operations, the cyber domain became one of most important aspects in strategic planning.

Society's dependence on this domain [1] has reached a point in which it is now considered as a new dimension of war, together with air, land and sea. In this new paradigm, a key aspect is to understand how actions performed in the cyber domain (space and time) affect the operations taking place in the other domains, so one can leverage actions in the cyber domain as tools to achieve the campaign objectives [2], [3]

Unfortunately, this is no trivial task, since it requires correlating cyber and physical behaviors in an integrated view that allows tasks to be evaluated in real time. The complexity embedded in this requirement implies, among other things, that an IT manager supporting critical infrastructures must be able to access all relevant data pertaining to the network and translate it to the support team in a way that allows them to understand the real impact of cyber threats to the network and what it means to the overall mission. Existing tools and methodologies cannot provide this level of information, and are not suitable to support complex cyber threat assessment in real situations. This is a major gap that to our knowledge has not been successfully filled, in spite of the relatively large body of research focused on the subject.

This paper addresses this gap by proposing a semantic framework that fuses physical and cyber data collected from existing sensors and retrieving information that is relevant to the assessment of cyber impact. It is designed to support analysts with an integrated view, one that correlates actions in the cyber domain with effects in other domains, allowing the evaluation of its impact on the operational objectives.

The proposed framework and its main aspects are illustrated and evaluated via a simulated air traffic scenario, which includes a large number of simulated flights.

This paper is organized as follows. Section II presents the main concepts necessary to understand the framework being proposed, as well as a sample of the most relevant approaches attained so far to address the problem. Section III describes the framework for evaluating the impact of a cyber attack on an operation occurring in the physical domain. The approach is discussed in Section IV, and illustrated with an analysis of a fictitious air traffic scenario build specifically to evaluate our research. Finally, Section V presents a few considerations and issues that must be addressed in future research aimed to improve the approach.

II. BACKGROUND AND RELATED RESEARCH

The main concept to present is *mission*. As discussed in [4], a mission is the task (or set of tasks), together with its (their) associated purpose, that clearly indicates the action to be taken assigned to an individual or unit.

Three other important concepts are *Situation Awareness*, *Impact Assessment* and *Threat Assessment*. The first, as described in [5], is the perception of the elements of the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status into the near future to enable decision superiority.

The second important concept, Impact Assessment, involves the task of estimating the effects on situations of planned or estimated/predicted actions by the participants, including interactions between action plans of multiple players [6].

The third and last concept, Threat Assessment, can be understood as an expression of intention to inflict evil, injury, or damage. The focus of threat analysis is to assess the likelihood of truly hostile actions and, if they were to occur, projected possible outcomes [6].

From a general perspective, the second and third concepts can be seen as being part of the first, but with a difference in their focus. More specifically, while impact assessment looks for an "internal" understanding (i.e., what is happening and why should I care?), threat assessment seeks the same understanding from the enemy's viewpoint (i.e., how they can hurt us). More important to our research is the fact they all these concepts imply a means to assess the mission. In other words, all must go through the process of specifying and maintaining a reasonable degree of confidence in mission success, which is linked to the concept of Mission Assurance [7].

Literature on the subject of measuring effectiveness of a mission points to two major approaches. The first is to use the concept of *task* as the evaluation basis, while the second instead focuses to the *effects* [8]. The framework presented in this paper adopts the second approach.

The main approach to provide mission understanding involves using a set of distributed sensors to detect intrusions and to uncover attack paths. The preliminary research on the subject is due to Denning [9] and Bass [10]. Schneier [11] proposed the use of an attack-tree to measure effect, which allows understanding of the relationships between attacks, as well as how one attack over a cyber asset affects other assets. In spite of the advances above cited, the problem of determining the impact of a cyber attack on a (mission) task still persists, since no methodology exists to effectively map cyber assets to tasks. Furthermore, these techniques are not capable of dealing with some common types of cyber attacks, rendering them unsuitable for impact assessment in the current state of the art in cyber warfare. For instance, when an attack is new (e.g. a zero-day attack), its signature is unknown and there will be no attack-tree associated with it. As a result, it will be extremely difficult to identify its attack pattern by the time it occurs.

The above limitation illustrates the need for new approaches. A more comprehensive one would involve identifying attacks, highlighting significant events and then understanding the importance of them in a system [12]. To assess the importance of events, one must understand how the process of planning and implementing a mission works. Topological Analysis of Network Vulnerability (TVA) [13] is meant to provide such understanding. TVA supports an analyst in measuring the impact of a threat through the evaluation of topological aspects of the environment. The main weakness of this approach is the absence of an explicit mapping between the mission and the infrastructure supporting it. As a result, this becomes yet another cognitive burden implicitly assigned to the analyst, a solution that clearly does not scale well with the increasing complexity of the operational environment.

Another related approach can be summarized by the work on Mission-Oriented Risk and Design Analysis (MORDA) [14] and on the Security Optimization Countermeasures Risk and Threat Evaluation System (SOCRATES) [15]. In this approach, all components that exist in the problem (mission, resources and threats) are mapped and used in the analysis. However, the mapping process is very complex and requires continuous iteration with the human analyst (i.e. human-in-the-loop), who needs to provide constant feedback and input to the methodology. As a consequence of its demand for human interaction, this approach tends to be applied in the planning phase, while being less suitable to the more time intensive environment found in real time decision making scenarios.

Another methodology that relates to the problem addressed in this paper is Cyber Mission Impact Assessment (CMIA) [7], [16]. CMIA presents a way to (manually) associate mission and infrastructure, and use the resulting association to support the assessment of mission assurance.

In a typical analytical process using CMIA, each attack is simulated and its associated impact is calculated. Then, all attacks and assets are correlated and the paths with the highest cost are prioritized. The major deficiency of this approach is its inability to evaluate more than one attack simultaneously, which prevents an assessment of the synergistic effect of coordinated attacks. This is a major liability, since in most cases the enemy would attempt to achieve an overall effect with parallel attacks that is much greater than the sum of the isolated effects of these same attacks.

The above mentioned works are a representative subset of current research related to evaluation of the impact of cyber threats, and can thus support the claim that the research problem remains unsolved. In summary, each approach suffers from in at least one of the two issues that can be singled out as the main causes for this situation. The first is the lack of a correlation (and, in some cases, computation) between the main components that are needed for impact assessment, the mission and its supporting infrastructure. The second cause for failures is the inability to provide real-time analysis of these two components and their interactions. The proposed framework is meant to address both, with a unique combination of semantic technologies, operations research, and simulation, which we explain in the next Section.

III. Evaluating the Impact of Cyber Threats

This paper proposes ARGUS, a new Framework that evaluate the impact of a cyber attack on a mission. ARGUS is comprised of four main phases: 1) modeling of mission, 2) modeling of network architecture, 3) collecting cyber and mission information, and 4) developing impact assessment. These phases are depicted in Figure 1.

As implied in the diagram, the core idea within ARGUS is to capture the mission and infrastructure information and consolidate it in an integrated data representation, which allows for a comprehensive analysis to be performed.

A. Modeling of Mission

The first phase in ARGUS involves modeling of mission, which is achieved by the use of a business process language.

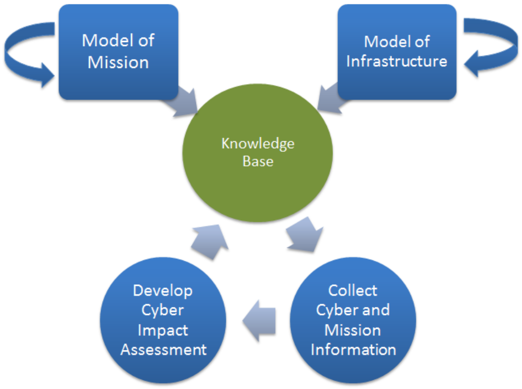

Model of Mission
Model of Infrastructure
Knowledge Base
Develop Cyber Impact Assessment
Collect Cyber and Mission Information

Figure 1. ARGUS major phases

The goal of this phase is to capture the most important information of the mission within the model. Importance here, of course, is measured with respect to its relevance to impact assessment, and includes the tasks, relationships between the tasks, objectives, resources required to develop the mission and, finally, performer (i.e., entity or set of entities that has the responsibility to perform the mission).

In our current research, we leveraged previous experience within our group and made the design decision of capturing these aspects using the Business Process Modeling Notation (BPMN) language [17]. However, any business modeling language with the ability to capture the information described above could have been used and, therefore, might be used with the framework in the future.

One of the most important features of the ARGUS is its reliance on semantic technologies to ensure consistency when used in multiple domains. Therefore, although a business modeling language is used as the basis for information elicitation (BPMN, in the current implementation of the ARGUS), all information captured is stored in an ontology-based information representation repository. The ontology supporting the repository was developed using the most recent version of the W3C recommended OWL 2 Web Ontology Language [18]. In fact, to illustrate the advantages of using an ontology-based framework, it should be emphasized that we didn't have to actually develop a mission ontology from scratch, but we simply imported and made some adaptations to existing work by others. That is, the ontology itself is an adaptation of the one defined in D'Amico et al. [19], while architecture is based on that of Mateus et al. [20].

In our context, the main concept in a mission is activity (see figure 2). An activity has a set of pre and post conditions and one goal. His goal is to produce one or more effects over a resource. An activity can be measure, enabling that can be understand the state of the mission's components.

Due to its main focus on business, BPMN lacks native support for some of the mission information that needed to be captured. Thus, we had to extend its basic structure to accommodate our representational requirements. Figures 2 and 3 illustrate some of the extended attributes (marked with a circle in the figures), which are present in the mission ontology supporting the repository.

The use of a business language (BPMN in the current implementation) was not only convenient as a development tool for the framework, but also proved to be rather suitable for capturing the main aspects of a mission, especially when it is used in civilian environments such as air traffic management, nuclear power plants, and others. Its business-oriented notation made it easier to accommodate the concepts of a mission in the Air Traffic Domain that we are using in the evaluation of the research, while also having a relatively straightforward mapping to the associated concepts in the mission ontology.

One example of a business-oriented concept being mapped to the mission ontology is that of a Pool. To model a mission, an analyst starts by describing the Organizations that participate in the process of accomplishing the mission. These can be squadrons, sectors, departments, battalions, or any functional structure involved with the mission details. Pool is the BPMN concept used to describe such organizations.

We expect the currently developed mapping to be relatively robust when applied along with the framework to other domains. Table I summarizes of the mapping developed in this initial phase of our research.

Table I
MAPPING BPMN TO THE MISSION ONTOLOGY

Concept Source	
Mission Model	**BPMN**
Organization	Pool
System	Lane
Activity	Task
Service	Performer
Condition	Gateway or Event

The ARGUS approach only builds mappings between automated processes, although BPMN is able to support non-automated ones. A service is understood as the entity responsible for performing tasks (activities), while a system is a collection of services. To ensure a proper correlation between business and infrastructure data, the analyst must describe where the service is provided, using his address and ports.

The framework supports the identification of relevant information from raw data captured by the sensors. In order for this to be accomplished, information regarding the effect, conditions and service level are described using rules. More specifically, an *effect* is the result, outcome, or consequence of an *action* (task) over a resource. Further, a *condition* can be understood as the state of the environment or of a situation in which a performer (service) performs or is disposed to perform an task. Finally, *service level* refers to the minimum (or maximum, depending on the requirement) standard that a service is expected to reach with confidence.

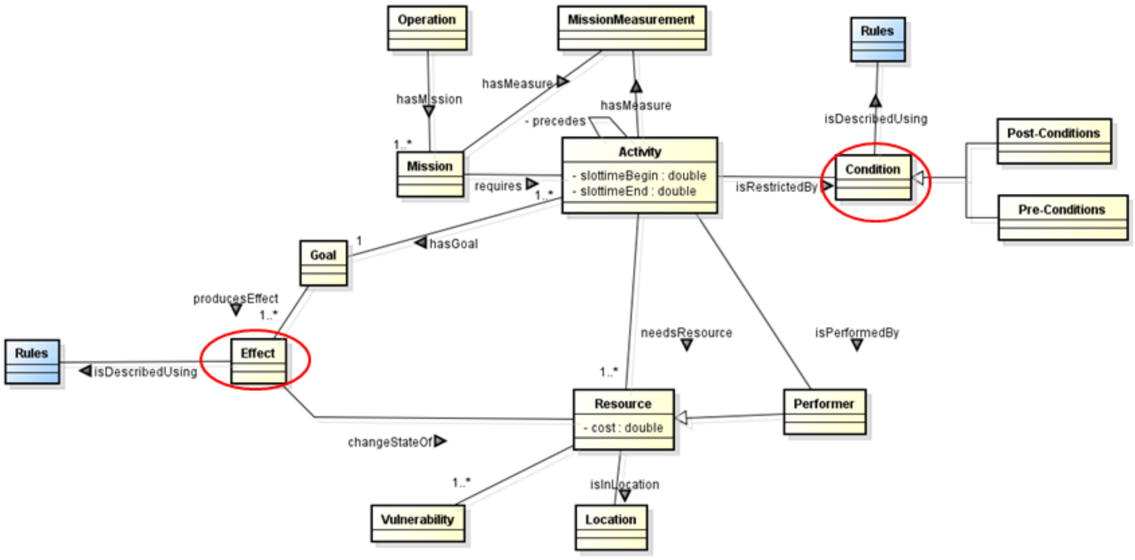


Figure 2. The Mission Ontology

B. *Modeling of Network Architecture*

The second phase in ARGUS, modeling of network architecture, is in fact performed almost in parallel with the first. In this phase, all information about the infrastructure is captured using Simple Network Management Protocol (SNMP) [21] and stored in the ontology-supported information representation repository. The main concept in the ontology used to represent the infrastructure is Cyber Asset, which is also depicted in Figure 3. Cyber Assets are responsible for to host one or more service (which is who performs the activities needed by the mission). Through services, ARGUS maps the infrastructure in mission and vice versa.

Another important concept from BPMN is that of a *performer*, which was mapped to the mission ontology as *service* (cf. Table I). In BPMN, the performer concept defines the resource that is responsible for an activity. It can be specified in

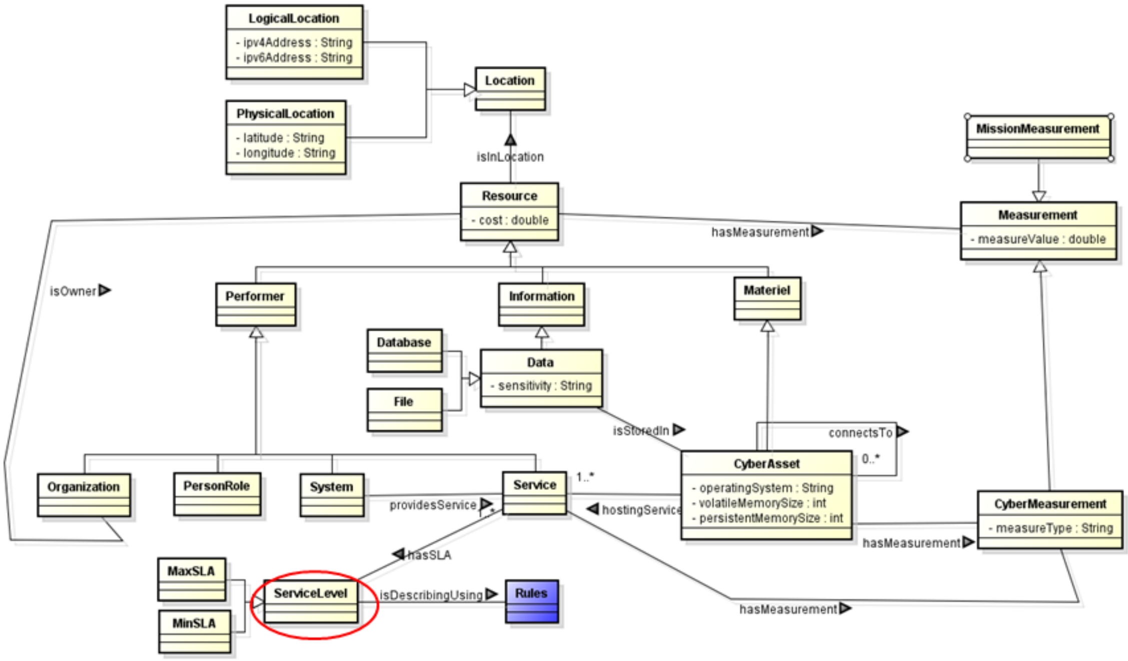


Figure 3. The Resource Ontology

the form of a specific individual, a group, an organization role or position, or an organization. Due to the above mentioned mapping, in ARGUS performers are services, which explains the need for analysts to specify the implementation address during the modeling. In other words, the correlation between the services and the cyber assets is made automatically by the framework via SNMP queries, which collect the UDP/TCP ports of the services via two tables residing in the Management Information Base (MIB) of each of the network hosts (*tcpConnLocalPort* and *udpLocalPort*).

To build the network archiecture and its variations, the framework performs queries on the other three tables residing in each host's MIB, the *ipRouteDest*, the *ipRouteMetric*, and the *ipRouteNextHop*. The combination of the information retrieved from these tables allows the Framework algorithm to infer the neighbors of the host, as well as the network distance between the host and nodes that were eventually discovered via the routing protocol embedded in the framework algorithm. Finally, the framework uses changes in those attributes (e.g. nodes added, nodes deleted, changes in nodes IP route metrics, etc.) as parameters for inferring the network dynamics. Besides the network information mentioned above, the framework also uses SNMP to retrieve a set of other infrastructure properties, such as memory (persistent and volatile) size, operating system, uptime, etc. It is outside the scope of this paper to explain in detail the framework algorithms and how each network parameter is assessed, more information on these details can be obtained from the work at the GMU/ITA C2 testbed (cf. [22]).

C. Collecting Cyber and Mission Information

The third phase in ARGUS involves the collection of relevant information. In this case, the criteria for information to be considered relevant is related to the value it adds to the overall understanding of the environment (i.e. how it improves situation awareness). This assessment is performed in accordance with the general scheme depicted in Figure 4.

The main concept in the scheme is *Situation*, which is an event or set of events that are meaningful to the mission. In ARGUS, events can be captured in any different ways. In our first implementation, we can retrieving the data existing in the SYSLOG Database [23] or by capturing network packets via a packet capture (PCAP) interface (e.g.through an intrusion detection system) [24]. Once an event is captured, the framework uses rules to classify it as being part of a situation. As previously mentioned, these rules will be applied to information retrieved from the network sensors and inserted into the framework through the BPMN's and Ontology's interfaces (cf. Figures 2, 3, and 4).

The design choice for describing the rules was the Semantic Web Rule Language (SWRL) [25]. SWRL extends a set of OWL axioms to include Horn-like rules, thus enabling Horn-like rules to be combined with an OWL knowledge base. The expressiveness achieved by this rule scheme is key to the framework's ability to capture aspects that cannot be easily captured using OWL, such as utilization of resources, mission requirements, and others.

Once all information needed from the business and infrastructure is retrieved, the events are captured from the sensors' input, and classified in accordance with relevant situations using rules. Then the framework is ready to evaluate the impact of the current state of the system on its main mission. In ARGUS, this evaluation is performed through four distinct types of analysis: *dependence paths*, *temporal*, *cost*, and *history degradation*.

The first type of analysis, *dependence paths*, aims to uncover problems in topology that have the potential to affect the accomplishment of the mission. The typical questions involved in this analysis include (but are not limited to) the following:

- In this state of the system, can the mission goal be reached?
- If task C fails, is there any path left to reach the goal?

The second type of analysis, *temporal*, seeks to define a window of interest in which the problem is solvable. The typical questions that are raised in this type of analysis include but are not limited to:

- What tasks need to be monitored at time T?
- How much time is needed to finish the task and accomplish its objective?

The third type of analysis, *cost*, is meant to identify when the cost starts to become a serious threat to the task execution. In other words, it evaluates the cost / benefit ratio of each task with respect to the overall mission. The typical questions to be answered in this analysis include:

- How much does this task cost?
- Do the benefits of this task justify the costs involved in its execution?
- If task C is compromised, does an alternative route have an acceptable cost?

The last type of analysis, *history degradation*, has the goal of understanding how fast the infrastructure is degrading. Its typical questions can be similar to the ones in each of the above tasks, but with a focus on the way the infrastructure assets are degrading and its associated impact on the overall mission.

D. Developing Impact Assessment

The fourth phase in ARGUS, *impact analysis*, is the main part of the framework. In order for this phase to be executed in real time, so the impact evaluation would be done as the mission unfolds, we have developed the reference implementation depicted in Figure 5.

The Cyber Situation Awareness engine (CyberSA Engine) is comprised of six modules. The first is the BPMN Module, which performs the tasks of getting mission information from a BPMN file, parsing it, and mapping the retrieved concepts to the mission ontology.

The SNMP and SYSLOG modules perform queries on all hosts and on the SYSLOG Server, respectively. When the associated answers are received, the module parses and converts them to the format they will be used in the system.

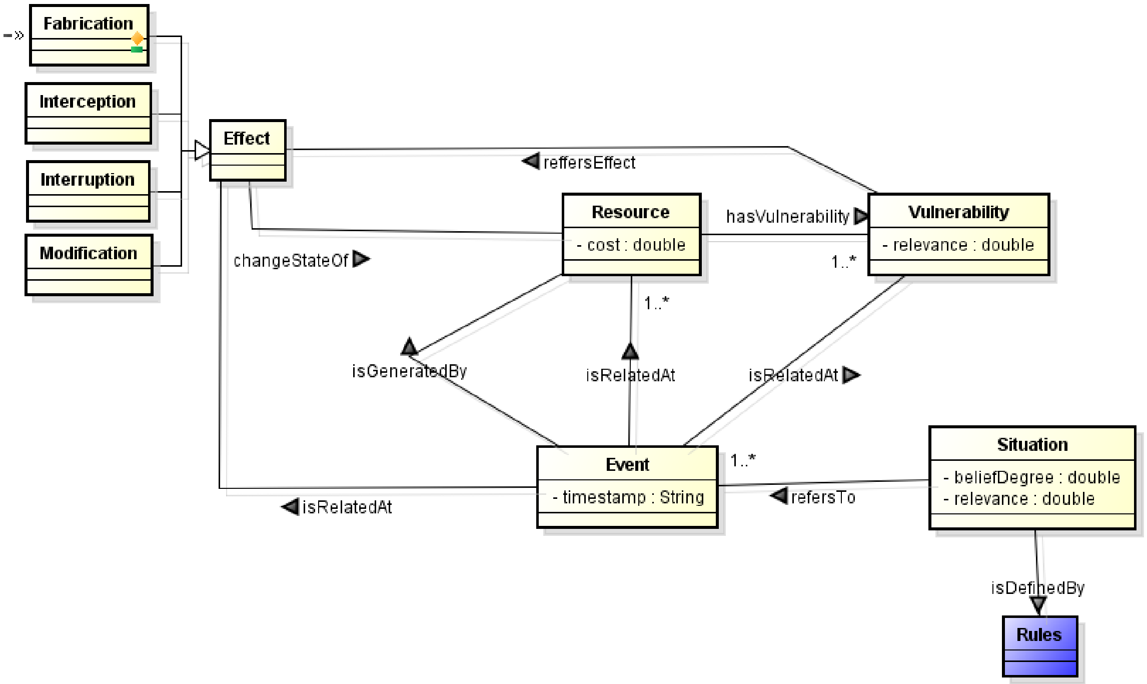


Figure 4. Capturing the Details of an Event

The PCAP module retrieves event data from the network. However, analysing the retrieved raw data is a time consuming and non-trivial task, so in our implementation we have made the design decision of using an external tool, TSHARK [26]. This tool is a terminal-oriented version of Wireshark designed for capturing and displaying packets when an interactive user interface is not necessary or not available. It has a set of filters that produces information in a format that is more readable to analysts.

Once the four modules above collect and process their respective information, the result needs to be made available in a consistent way so the CyberSA Engine can provide it to the users. This consistency is also achieved with the support of semantic technologies, via the implementation of a Semantic Fusion Module. The main services this module provides are making inferences and applying rules, which were written by analysts using the GUI.

The Semantic Fusion Module uses two libraries to provide its features. The first is the OWL-API [27], a Java API and reference implementation for creating, manipulating and serializing OWL Ontologies. The second is Pellet [28], which is an OWL 2 reasoner that provides standard and cutting-edge

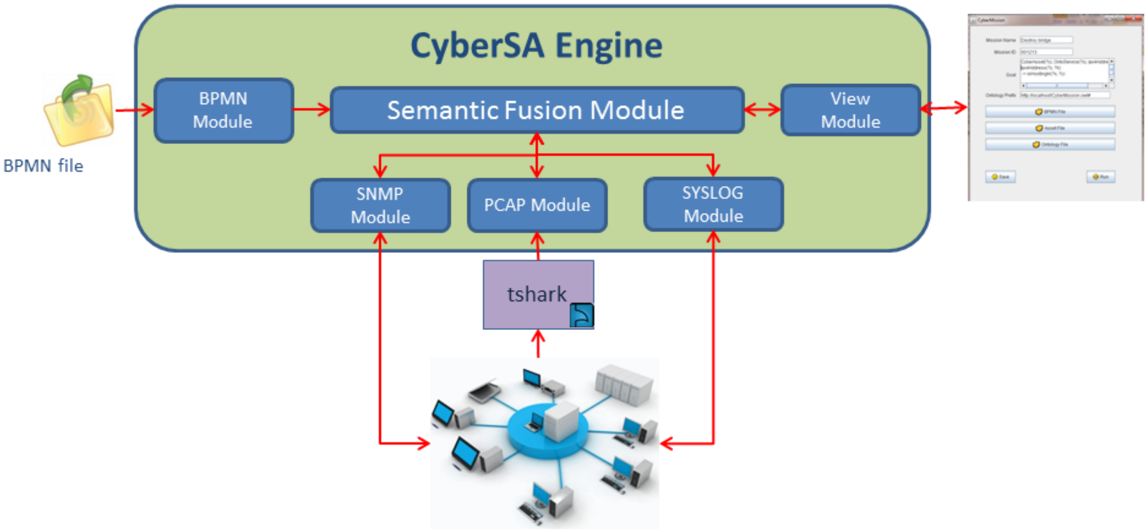
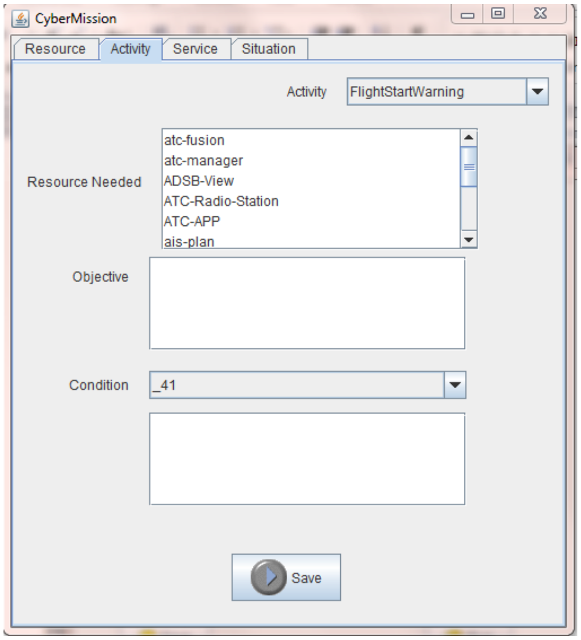


Figure 5. The CyberSA Engine

reasoning services for OWL ontologies.

The last module of the CyberSA Engine is the View Module, which provides the interface to analysts. The main goals of this interface are to allow analysts to provide information the system cannot obtain automatically, and to write the rules used by the system's inference engine.

Figure 6 is an example of a typical form of the system's GUI, in this case one that allows the analyst to setup a task. In the combo box depicted in the figure (named as "Activity"), the analyst chooses the type of activity he wants to set, as well as the associated fields - which are shown in a contextual fashion with support from the mission ontology. In the example, the analyst chose the activity "FlightStartWarning", and was then presented with three fields. In the first field, the analyst is presented with the resources that he needs to do the task. In the remaining two fields, the analyst is expected to describe, using rules in SWRL syntax, how to measure the task progress and the conditions this measure will be performed.



Activity: FlightStartWarning

Resource Needed:
atc-fusion
atc-manager
ADSB-View
ATC-Radio-Station
ATC-APP
ais-plan

Objective:

Condition: _41

Figure 6. The ARGUS User Interface

By means of this GUI, the system will guide the analyst through a process in which he will be able to define the activity, the cost of resources, the service's SLA, and other rules that must be defined given the relevant situations. The View Module also provides classification of the event (i.e. the situation(s) it pertains to).

IV. Discussion

A simulation of an air traffic scenario was developed to evaluate the framework, verifying its ability to generate the relevant situation assessment and present it to the analyst. The simulation is based on a real scenario, located at the Campos basin in Brazil, where a heavy helicopter operation is held to support maritime oil platforms sixty to eighty miles offshore. The mission described in this scenario thus involves air traffic service where the aircraft consume the smaller amount of fuel and the system generates a low number of collision resolution events. A collision resolution event happens when two aircraft fly within a distance (vertical or horizontal) that is smaller than the safety rules defined by law.

The simulation includes three distinct air traffic services organizations (cf. Figure 7). The first is the AIS (Aeronautical Information Service), which has the responsibilities of inserting the flight plan into the system and getting all clearance necessary for the aircraft to fly. The second service modeled is the Radio Station, which gets information on flight tracks (i.e. aircraft) within its area of coverage and sends it to the APP (Ground-controlled Approach) Service. Finally, the APP service performs three main tasks: fuse track information, present it in a controller view and generate alerts to be used by a monitoring system.

The simulation was developed using the C2 Simulation Testbed [22], a joint project between the C4I Center at George Mason University (GMU) and the C2 Lab at the Instituto Tecnológico de Aeronáutica (ITA) in Brazil. The testbed allows the emulation of any infrastructure behavior and the simulation of all aspects of the physical environment (aircraft flights, collisions, etc). The current evaluation scenario includes fourteen aircraft that take off from three different airports and go to the oil platforms. The flight plan was developed to generate collision warnings, allowing the framework to generate situations of interest. A view of this scenario using the C2 Simulation Testbed is presented in Figure 7.

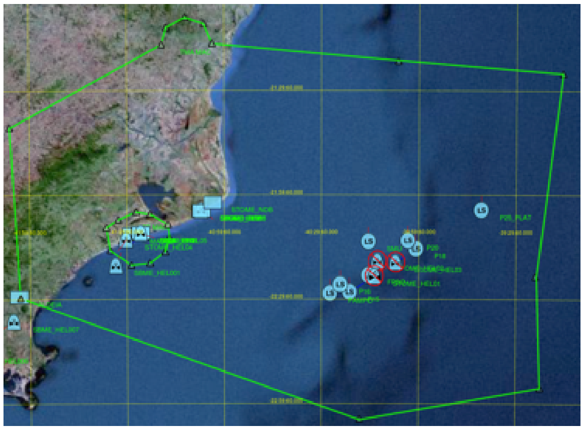


Figure 7. The Simulation in VRForces

A major aspect that is needed for the framework to define relevant situations is the proper definition of the rules by analysts. Among other things, these rules formally establish to the system the conditions that restrict the task, the goal of mission in general, the objective of each task, and other aspects that are important in filtering the raw data coming from the sensors. In addition to these aspects, another key use of rules is to create relations that are not explicit in the domain. As an example, the link between cyber assets and services can be defined by this simple rule:

$CyberAsset(?y), OntoService(?x), ipv4Address(?x, ?k),$ $ipv4Address(?y, ?k) \rightarrow isHostingIn(?x, ?y)$. Therefore, it is fair to say that the combination of SWRL rules and OWL 2 statements to link the physical and cyber domains is at the heart of the system's goal of evaluating mission impact.

V. Future Research

This paper presented an approach for connecting the cyber and physical domains, with the objective of assessing the impact that actions in the former have in the latter. This is research in progress in an area where clear answers are usually not attainable, mostly due to the complexity as well as to the level of subjectivity involved in real time impact assessment. As such, the framework presented here should be seen as a first step of a steep ladder. Yet, it is a firm step, since after attempting various approaches we remain convinced that the solution to this problem relies in a combination of techniques where semantic technologies and simulation play a major role.

The software modules, including the ontology and some of the rules, that together comprise the framework are already implemented, and we are currently evaluating its performance via the C2 Simulation Testbed. Preliminary results are promising and should be available soon. Our future work path includes aspects such as the usability of the system, and others that rely on semantic technologies to alleviate the reliance on analysts to provide domain knowledge in the form of SWRL rules.

Acknowledgment

The authors recognize VT MÄK and Scalable Network Technologies for providing the tools and support needed to develop the Testbed. Our gratitude is also extended to Latin-Media SA, who provided support for the testbed in Brazil.

References

[1] G. Eason, B. Noble, and I. N. Sneddon, "On Certain Integrals of Lipschitz-Hankel Type Involving Products of Bessel Functions," *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, vol. 247, no. 935, pp. 529–551, Apr. 1955. [Online]. Available: http://rsta.royalsocietypublishing.org/content/247/935/529

[2] M. Endsley, "The application of human factors to the development of expert system for advanced cockpits." in *Annual Meeting of Human Factors and Ergonomics Society*. Human Factors Society, 1987, pp. 1388–1392.

[3] J. Boyd, "OODA Loop." Center for Defense Information, Tech. Rep., 1995.

[4] DoD, *DODAF. DoD Architecture Framework Version 2.0 - Volume 1: Introduction, Overview, and Concepts.*, DoD Std., 2009.

[5] J. Salerno, M. Hinman, and D. Boulware, "A situation awareness model applied to multiple domains," in *Proceedings of SPIE*, vol. 5813, 2005, p. 65.

[6] E. Bosse, J. Roy, and S. Wark, *Concepts, models, and tools for information fusion*, A. House, Ed. Artech House, 2007 2007.

[7] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "A systems engineering approach for crown jewels estimation and mission assurance decision making." in *IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, 2011.

[8] M. J. Fiebrandt, C. Mills, and T. Beach., "Modeling and simulation in the analysis of a joint test and evaluation methodology," in *Spring Simulation Multiconference*, vol. 3. Society for Computer Simulation International, 2007, pp. 251–256.

[9] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. 13, pp. 222–232, 1987.

[10] T. Bass, "Multisensor Data Fusion for Next Generation Distributed Intrusion Detection Systems," in *IRIS National Symposion*, 1999.

[11] B. Schneier, "Attack trees: Modeling security threats," Dr. Dobb's journal, December 1999.

[12] O. S. Saydjari, "Cyber defense: Art to Science." *Communications of the ACM - Homeland Security*, vol. 47, no. 3, March 2004.

[13] S. Jajodia, S. Noel, and B. O'Berry, "Topological Analysis of Network Attack Vulnerability." *Managing Cyber Threats*, vol. 5, pp. 247–266, 2005.

[14] S. Evans, D. Heinbuch, E. Kyle, J. Piorkowski, and J. Wallner, "Risk-based systems security engineering: stopping attacks with intention," *IEEE Security and Privacy*, vol. 2, pp. 59–62, 2004.

[15] D. L. Buckshaw, G. S. Parnell, W. L. Unkenholz, D. L. Parks, J. M. Wallner, and O. S. Saydjari, "Mission Oriented Risk and Design Analysis of Critical Information Systems," *Military Operations Research*, vol. 2, pp. 19–38, 2005.

[16] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "Computing the impact of cyber attacks on complex missions." in *2011 IEEE International Systems Conference (SysCon)*, 2011, pp. 46–51.

[17] OMG, *Business Process Model and Notation (BPMN) 2.0*, http://www.omg.org/spec/BPMN/2.0, OMG Std., 2011.

[18] W3C, *OWL 2 Web Ontology Language*, http://www.w3.org/TR/owl2-overview/, W3C Std., October 2009.

[19] A. D'Amico, L. Buchanan, J. Goodall, and P. Walczak, "Mission Impact of Cyber Events: Scenarios and Ontology to Express the Relationships between Cyber Assets, Missions, and Users." AFRL/RIEF, Tech. Rep. OMB No. 0704-0188, December 2009.

[20] C. J. Matheus, M. M. Kokar, K. Baclawski, J. A. Letkowski, C. Call, M. Hinman, J. Salerno, and D. Boulware, "SAWA: An assistant for higher-level fusion and situation awareness," *Proceedings of SPIE*, vol. 5813, no. 1, pp. 75–85, 2006. [Online]. Available: http://link.aip.org/link/?PSI/5813/75/1&Agg=doi

[21] J. Case, M. Fedor, M. Schoffstall, and J. Davin, *A Simple Network Management Protocol (SNMP)*, The Internet Engineering Task Force (IETF) Std. RFC 1157, May 1990. [Online]. Available: http://www.ietf.org/rfc/rfc1157.txt

[22] A. B. Barreto, M. Hieb, and E. T. Yano, "Developing a Complex Simulation Environment for Evaluating Cyber Attacks," in *I/ITSEC*. I/ITSEC, 2012, will be published in I/ITSEC 2012.

[23] R. Gerhards, *The Syslog Protocol*, http://tools.ietf.org/html/rfc5424, IETF Std. rfc5424, March 2009.

[24] E. Nemeth, G. Snyder, S. Seebass, and T. Hein, *UNIX System Administration Handbook*. Prentice Hall, 2000.

[25] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML," http://www.w3.org/Submission/SWRL/, W3C Member Submission, May 2004.

[26] Wireshark, "Wireshark," http://www.wireshark.org/, 2012.

[27] M. Horridge and S. Bechhofer., "The OWL API: A Java API for OWL Ontologies." *Semantic Web Journal 2(1), Special Issue on Semantic Web Tools and Systems,*, pp. 11–21, 2011.

[28] "Pellet: OWL 2 Reasoner for Java," http://clarkparsia.com/pellet/, 2012.