THEMIS: Threat Evaluation Metamodel for Information Systems

Csilla Farkas[1] (farkas@cse.sc.edu)
Thomas C. Wingfield[2] (twingfield@potomacinstitute.org)
James B. Michael[3] (bmichael@nps.edu)
Duminda Wijesekera[4] (dwijesek@gmu.edu)

[1] Dept. of Computer Science and Engineering, USC, Columbia, S.C. 29208
[2] The Potomac Institute for Policy Studies, Arlington, Va. 22203
[3] Dept. of Computer Science, Naval Postgraduate School, Monterey, Calif. 93943
[4] Dept. of Information and Software Engineering, GMU, Fairfax Va. 22030

Abstract. THEMIS (**Th**reat **E**valuation **M**etamodel for **I**nformation **S**ystems) is a description logic-based framework to apply state, federal, and international law to reason about the intent of computer network attacks with respect to collateral consequences. It can be used by law enforcement agencies and prosecutors to build legally credible arguments, and by network designers to keep their defensive and retaliatory measures within lawful limits. THEMIS automates known quantitative measures of characterizing attacks, weighs their potential impact, and places them in appropriate legal compartments. From the perspective of computer networks, we develop representations and a way to reason about the non-network related consequences of complex attacks from their atomic counterparts. From the perspective of law, we propose the development of interoperable ontologies and rules that represent concepts and restrictions of heterogeneous legal domains. The two perspectives are woven together in THEMIS using description logic to reason about and guide defensive, offensive, and prosecutorial actions.

1 Introduction

Conventional security measures are designed to prevent attacks and to discourage attackers from continuing their attacks. Once an intrusion into a system or misuse of the a system is detected, the system response is to disengage itself from the rogue processes performing the intrusion or misuse actions. Current efforts aim to capture behavioral information about malicious users that can be used as the basis for developing sophisticated responses to intrusions, and to establish guilt of supposed intruders. Software and hardware decoy systems have been developed [12, 11, 16, 5] that allow observation and recording of attackers' activities as a form of counterintelligence in support of countermeasures.

In addition to the decoy systems, various organizations collect cyber crime related data, ranging from system log records to outputs of sophisticated intrusion detection systems. However, only a few attempts have been made to

develop an integrated framework to analyze these data [1] and to incorporate legal restrictions [15, 13] regarding data collection, dissemination, usability, and response [23].

In this paper we focus on the specific problem of evaluating the "intent" and the consequences of computer network attacks from the perspective of national security. This evaluation is especially important to justify retaliatory actions and ensure their lawfulness while providing maximal defense against an adversary. Judging attackers' intent and evaluating the consequences of the attacks are difficult tasks because both the attackers and the consequences of attacks can be categorized into diverse domains. For example, an attacker may be a "script kiddie," a hacker, or a member of a cyber terrorist group, and the consequences (effects) of a computer attack may range from mere inconvenience to human injuries and threats to national security. Coincidently, judging intent requires system-level evidence to be collected, assimilated and maintained to satisfy legal definitions and standards of acceptability, non tampering, and non-repudiation. We propose an approach that relies on legal guidelines to evaluate computer attacks.

With respect to determining intent and damage caused by cyber attacks, from a legal perspective, Schmitt [18] and, subsequently, Wingfield [23] propose seven factors to distinguish between economic and political coercion (non-armed) and armed coercion. These factors (explained in detail in section 2.1) are: severity, immediacy, directness, invasiveness, measurability, presumptive legitimacy, and responsibility. Evaluation of cyber attacks, using these factors, is referred to as the Schmitt Analysis. Michael et al. [14] demonstrate, via a case study of kinetic and cyber attacks on a safety-critical software-intensive system, the applicability of the Schmitt Analysis to determine whether an attack has risen to the level of "use of force" under international law. However, subjective and ad-hoc evaluation of the Schmitt factors can be erroneous with widely varying results, and may not be able to handle large-scale, distributed attacks. It is necessary, therefore, to develop systematic and automated methods for Schmitt Analysis.

This paper presents our current results in developing a metamodel to evaluate characteristics of computer network attacks, and to reason about legal aspects of these attacks and their responses. In particular, we propose a formalism to represent the characteristics of an attack as stated in [14, 18, 23] so that they can be effectively used in an automated reasoning system.

We present a rudimentary policy specification language using the Rule Markup Language [2] syntax. Rules are developed to express legal and military constraints, and the evaluation criteria for the Schmitt Analysis. Reasoning based on these rules will allow the classification of an attack (or coordinated attacks) and support the formulation of counter actions. Our aim is to develop THEMIS using the latest Semantic Web standards such as SWRL (A Semantic Web Rule Language combining OWL and RuleML [4]). These techniques are shown to be promising to support interoperation and reasoning over heterogeneous data domains.

The organization of the paper is as follows. Section 2 contains background information. Section 3 contains the overview of the THEMIS framework. Description of the components of THEMIS are presented in Section 4, where Section 4.1 describes the ontology representation, Section 4.2 the policy specification, and Section 4.3 the conflict resolution and default strategy. Finally, we conclude in Section 5.

2 Background information

2.1 Schmitt Analysis

Factors, to distinguish between military (armed), and economic and political coercion of computer network attacks, were identified by Schmitt [18] and Wingfield [23]. Analysis of computer network attacks based on these factors results in justifications that are considered *use of force* under Article 2(4) of the United Nation's Charter. In the perspective of increased attacks and state sponsored attacks against national resources, the Schmitt analysis represents a crucial step to ensure preservation of international law while allowing maximal level of defense. Details of the factors are:

Severity: Measure of physical damage and human casualties. For example, value of compromised resources, human injuries, size of the affected area, etc.

Immediacy: Time period before the effect is observed, and how long it lasts. For example, effects can be observed (occurred) minutes after the attack, system is unavailable for a week, etc.

Directness: Whether the attack caused the results directly (e.g., DDOS attack against Navy communication system) or indirectly (e.g., disabling the Internet to support attack on a particular system).

Invasiveness: Did the attack require crossing national boundaries?

Measurability: Can the characteristics (damage) of the attack be easily measured? (e.g., physical or financial damage, lost system time, etc.)

Presumptive Legitimacy: Is the action analogous to actions (such as large scale application of violence) that could only be presumed legitimate if undertaken by a state?

Responsibility: Is a nation-state responsible, and is it publicly stating this?

2.2 Semantic Web

The need for interoperation and reasoning over heterogeneous data and knowledge sources is not unique to our application area. The WWW Consortium is developing a standard referent known as the *Semantic Web* for which humans specify the rules of engagement in their own language which in turn will be automatically translated into machine-executable instructions for exchanging self-describing information. Our framework conforms to the existing technologies on the Semantic Web and can be fully integrated with them.

The Semantic Web [7] is envisioned as several, interconnected layers. The lowest layer (XML and namespace) consists of machine-readable code. The next layers, RDF and ontologies, consists of resource definitions available at the machine level. The level on top of the ontologies consists of rules. The top layers consist of logical reasoning engines that would assist in constructing proofs. The main lesson learned from the Semantic Web is that the ontologies provide the crucial translation between applications within a domain.

Using Semantic Web [3] technologies seem to be promising for a variety of domains, like e-business rules [10], intrusion detection [20], and agent-based systems [9]. Most of our proposed legal formulations relate to the ontology and rules layers. For the framework to function, a representative of each legal jurisdiction must publish its own ontologies and rules, making them available for viewing by other communities. For example, if the Federal Bureau of Investigation (FBI) is investigating a cyber attack launched from a site in the European Union (EU), it could consult the EU rules repository to determine the extent of permissible legal countermeasures. By using the ontology layer, different law enforcement agencies could publish their own conceptualization (ontologies), so that they could be seamlessly compiled to fight anticipated cyber crimes.

2.3 Legal Reasoning

Developing legal reasoning models is not new. With the increased computational power and the expressiveness of the representational techniques several reasoning models have been developed. Most contributions are based on artificial intelligence. A difficult problem here is to handle so called *open-textured* concepts, i.e., lack of clear definition of legal concepts, requiring experience and common sense to determine their applicability. Nevertheless, several models have been developed. The main categorization of these models are rule-based (logical), case-based, and adversarial reasoning.

Case-based reasoning models rely on previous experiences, maintained as distinct cases, to analyze current cases. Examples of case-based legal reasoning systems are HYPO [17, 6] and GREBE [8]. Some of the systems combine rule-based and case-based methods, like the CABARET [19] system. The two methods are also used for legal argumentation.

A relatively new area of legal reasoning research is to develop ontologies to represent conceptualization of legal domains. Visser et al. [21, 22] give an overview of the roles ontologies may play in the legal contexts. These roles include knowledge sharing, verification of knowledge base, software engineering, knowledge acquisition, knowledge reuse, and domain-theory development. However, none of these works address the legal aspects of cyber attacks or the assessment of the attack consequences from the legal perspective.

3 THEMIS Framework

Cyber activities are rarely contained within geographical borders, therefore the identification of presumed violators and preservation of legal evidence require

coordinated efforts in multiple jurisdictions, each of which has its own set of legal standards. To support this coordinated effort, it is necessary to enable knowledge sharing and reasoning over diverse knowledge bases. Ontologies, developed by each jurisdiction, support this need by giving explicit specification of the concepts and the relationships assumed to exist among them. In addition to the local ontologies, we propose the development of meta-ontologies which relate various local ontologies. This high-level representation allows the user to compare, analyze, and reason with the legal domain of the different jurisdictions.

Concepts and their relationships, provided by the ontologies, are captured by the formal syntax of the Attack Response Policy (ARP) specification language. The ARP language is a logical language allowing to express legal and response policies. For this, we define the precise alphabet for ARP and a set of predicates (originating from the ontologies) to create policies for legal reasoning, argumentation, attack evaluation, and response.

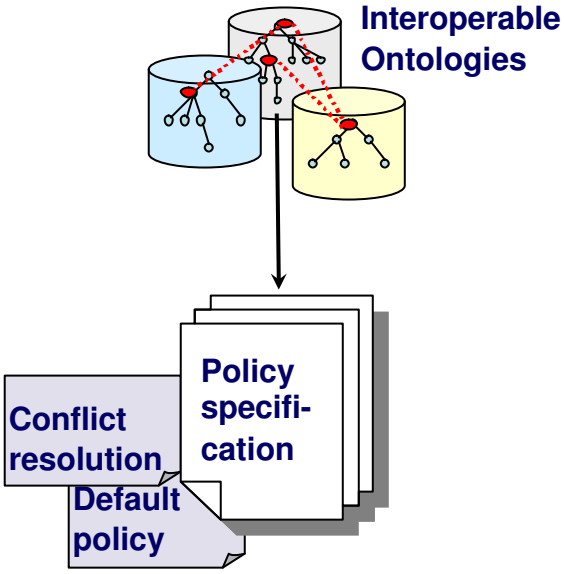

Fig. 1. Components of the THEMIS Framework

The two main components of THEMIS are: 1) Legal, military, and computer network attack concept ontologies, and 2) the ARP specification language, capable of reasoning with these ontologies. In addition, THEMIS is capable of resolving conflicts and applying default policies. Figure 1 shows the components of THEMIS.

3.1 Network Attack Example

The complexity of formulating the proper rule of law, deconflicting areas of overlap with other such rules, and correctly applying it to the facts at hand is a subtle and complex process. However, it is possible to construct several general principles of application. One such principle is that for any given problem, there is a hierarchy of laws which prevents a logical impasse with apparently conflicting laws. It is not always clear, but almost always possible to establish such a priority in the application of multiple rules.

Consider the following example showing the cascading nature of the network attacks, and the legal restrictions governing the investigation. Assume, that the computer system of the air traffic control tower of Airport `XX` in state `X` is attacked, causing the system to malfunction. Due to erroneous information, two airplanes crash, causing physical damage to the planes and the nearby buildings, and loss of human life.

Investigators in state `X` would be permitted to use all the means at their disposal to identify, locate, and apprehend the perpetrators because the intrusion

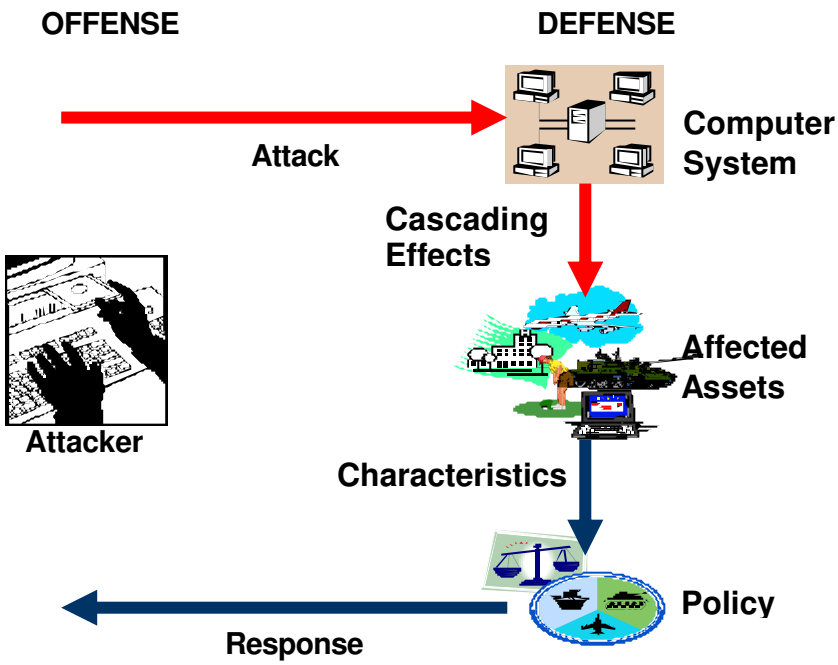

Fig. 2. Attack Analysis Model

to a computer system took place in state `X`. Assume, that the investigation of the air traffic system reveals that the network attack originated from state `Y`. This requires to acquire the appropriate permissions from the jurisdiction of state `Y`. Without evidence of the attack origin and permissions, a violation of the US Constitution (e.g., an unreasonable search or seizure, prohibited by the Fourth Amendment and extended to the states through the Fourteenth Amendment) would occur. This priority of law (federal constitutional, federal legislative, state constitutional, state legislative) establishes a priority among the applicable rules.

4 THEMIS Components

An overview of the two main modules is given in this section, using the Schmitt Analysis as a representative example. Figure 2 shows the workflow of the attack analysis and response.

4.1 Ontology Representation

There are several ongoing efforts to develop ontologies for cyber attacks [20, 9]. However, most of these works focus on the technical aspects of the attack (e.g., type of attack, system components exploited by the attacks, means of exploitation) and are limited in describing the legal characteristics of the attacks and their non-network related cascading consequences. To strengthen security and to provide efficient and lawful response it is necessary to evaluate auxiliary consequences that may reach out of the domain of computer systems, and develop measurements of the damage caused by cascading effects.

Our ontology representation is compatible with current work but extends it in four respects: 1) provides measurements of damage assessment, 2) represents cascading and escalating effects of cyber attacks, 3) represents legal concepts, and 4) provides metadata necessary for distributed legal reasoning. Figure 3 shows the graphical representation of computer network attacks from the perspective of consequences. Figure 4 shows a sample of the corresponding OWL representation.

Figure 5 shows the relationship between the consequences and the factors used in the Schmitt Analysis. Figure 6 shows a sample of the corresponding OWL representation.

4.2 Attack Response Policy (ARP)

To determine the lawfulness of the responses to computer attacks, we need to evaluate the direct and indirect damages caused by the attacks and generate policies for handling them. This includes damage to the target system (usually a computer network and its components) and the cascading effects of the compromised assets, caused by limited availability, integrity, or confidentiality. Cascading effects of an attack may reach out of the cyber domain, resulting in damage to physical and human assets. Attack consequences are categorized

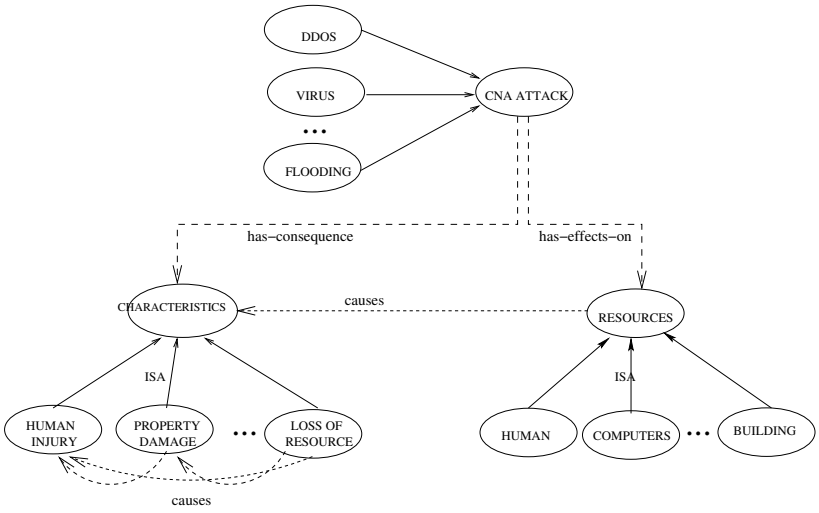

Fig. 3. Computer network attacks and their characteristics

according to the levels and types of the incurred damages. Note, that "attack consequences" refer to non-computational consequences (i.e., different from logical consequence). They refer to the results of the attacks in the target system, and their cascading and escalating effects. Given the aforementioned requirements, our goal is to develop language for specifying policy and law governing attack assessment and response. Technically, law governs human and automated-system behavior, whereas policy defines the latitude with which the law can be applied. In this paper, however, we use the term "policy," to encompass both law and policy as defined above. Ontologies supply the intra- and interdependencies of the affected assets. We are developing the basic alphabet and predicates for our Attack Response Policy (ARP) specification language.

We are planning to use SWRL [4] rules to reason about the attacks, their characteristics, and legality of responses. For the Schmitt Analysis, THEMIS reaches a classification of the attacks into three categories: "use of force," "arguable use of force," or "not a use of force" according to Article 2(4) of the UN Charter.

Attack Response Policy (ARP) Specification Language Our goal is to develop a set of symbols that can be used to build rules expressing policy and assessment requirements. The ARP specification language defines basic alphabet and predicate symbols. The alphabet of ARP language includes constant symbols, variables, functions, and terms. Constant symbols correspond to elementary concepts of the ontologies, like *computer, human, attack, etc.*. Variables range over the domain of these concept. Function symbols map related concepts, determine quantitative relations, and used to construct terms.

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rss="http://purl.org/rss/1.0/"
    xmlns="http://a.com/ontology#"
    xmlns:jms="http://jena.hpl.hp.com/2003/08/jms#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#"
    xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
  xml:base="http://a.com/ontology">
  <owl:Class rdf:ID="CNAATTACK"/>
  <owl:Class rdf:ID="CHARACTERISTICS"/>
  <owl:Class rdf:ID="RESOURCES"/>
  <owl:Class rdf:ID="DDOS">
    <rdfs:subClassOf rdf:resource="#CNAATTACK"/>
  </owl:Class>
  <owl:Class rdf:ID="VIRUS">
    <rdfs:subClassOf rdf:resource="#CNAATTACK"/>
  </owl:Class>
  <owl:Class rdf:ID="FLOODING">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="CNAATTACK"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="IMMEDIACY"/>
  <owl:Class rdf:ID="MEDICALCOST">
    <rdfs:subClassOf rdf:resource="#CHARACTERISTICS"/>
  </owl:Class>
  <owl:Class rdf:ID="IMMEDIACY"/>
  <owl:Class rdf:ID="MEDICALCOST">
    <rdfs:subClassOf rdf:resource="#CHARACTERISTICS"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="hasConsequence">
    <rdfs:range rdf:resource="#CHARACTERISTICS"/>
    <rdfs:comment>Relating attacks to characteristics</rdfs:comment>
    <rdfs:domain rdf:resource="#CNAATTACK"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasEffectsOn">
    <rdfs:domain rdf:resource="#CNAATTACK"/>
    <rdfs:range rdf:resource="#RESOURCES"/>
    <rdfs:comment>Attacks having effects on Resources</rdfs:comment>
  </owl:ObjectProperty>
    ...
```

Fig. 4. OWL representation of computer network attacks and their characteristics

We also define predicates, necessary to build rules expressing attack assessment requirements, legal and response policies. For example, the 4-ary predicate

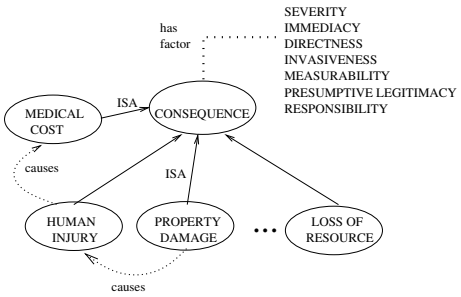

Fig. 5. Attack characteristics

symbol *attack* has the following form:

$$attack(a-id, a-name, orig, targ)$$

where the first argument is an attack identification number, the second is the name of the attack, the third is the originator of the attack, and the fourth argument is the target of the attack. For example, the fact that a distributed denial-of-service (DDOS) attack was launched from IP address 129.167.2.5 against IP address 168.145.21.26, and the attack was detected and numbered with attack identification number 29 can be represented with the ground fact (i.e., predicate with constant symbols only)

$$attack(29, DDOS, 129.167.2.5, 168.145.21.26)$$

The consequence predicate is a 3-ary predicate, with arguments attack ID, consequence type, and target.

$$consequence(a-id, c-type, targ)$$

Each Schmitt factor is represented as a predicate, with attack ID, target, and value arguments. For example, the predicate corresponding to factor *severity* is

$$severity(a-id, targ, val)$$

The 4-ary predicate *causes* captures dependencies between cascading consequences

$$causes(c-type_1, targ_1, c-type_2, targ_2)$$

where $c-type_i$ corresponds to the consequence type on the asset $targ_i$ $(i = 1, 2)$.

Policy Rules Using the predicates of the ARP language, we can build rules to reason about the damages, express legal restrictions, and determine legitimacy of counter actions. Each rule has the following form:

$$head \leftarrow body_1,\ body_2,\ \ldots\ ,\ body_n$$

```
…
<owl:Class rdf:ID="CHARACTERISTICS">
  <owl:Class rdf:ID="LOSSOFRESOURCE">
    <rdfs:subClassOf rdf:resource="#CHARACTERISTICS"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="hasFactor">
    <rdfs:comment>Relating characteristics with their factors</rdfs:comment>
    <rdfs:domain rdf:resource="#CHARACTERISTICS"/>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#SEVERITY"/>
          <owl:Class rdf:about="#IMMEDIACY"/>
          <owl:Class rdf:about="#DIRECTNESS"/>
          <owl:Class rdf:about="#INVASIVENESS"/>
          <owl:Class rdf:about="#MEASURABILITY"/>
          <owl:Class rdf:about="#RESPONSIBILITY"/>
          <owl:Class rdf:about="#PRESUMPTIVELEGITIMACY"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="causes">
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#HUMANINJURY"/>
          <owl:Class rdf:about="#PROPERTYDAMAGE"/>
          <owl:Class rdf:about="#LOSSOFRESOURCE"/>
          <owl:Class rdf:about="#RESOURCES"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:comment>Relation between different kinds of
characteristics</rdfs:comment>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#MEDICALCOST"/>
          <owl:Class rdf:about="#HUMANINJURY"/>
          <owl:Class rdf:about="#PROPERTYDAMAGE"/>
          <owl:Class rdf:about="#CHARACTERISTICS"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
…
```

Fig. 6. OWL representation of attack characteristics

where $body_1$, $body_2$, ... , $body_n$ is referred to as the body of the rule. Intuitively, if there is a valuation to the symbols of the *body* such that all predicates are true, then the head must also be true. We restrict our rules to use only the symbols of the ARP language, atomic datatypes, and ARP predicates, and to be bounded, that is, all symbols of the head must appear in the body.

The following example shows the rules to derive cascading consequences of an attack.

$$attack(a-id, a-name, orig, targ_1) \leftarrow attack(a-id, a-name, orig, targ), \\ consequence(a-id, c-type, targ), \\ causes(c-type, targ, c-type_1, targ_1)$$

$$consequence(a-id, c-type_1, targ_1) \leftarrow attack(a-id, a-name, orig, targ), \\ consequence(a-id, c-type, targ), \\ causes(c-type, targ, c-type_1, targ_1)$$

The two rules state that if there is an attack on the *targ* asset, and the consequence *c-type* of the attack causes a consequence $c\text{-}type_1$ on asset $targ_1$, then the attack is considered an attack on $targ_1$ and the consequence of the attack is $c\text{-}type_1$.

Rules to evaluate attacks from the perspective of the Schmitt Analysis can be created in a similar fashion. Predicate, generated for each factor, are used in rules implementing the Schmitt Analysis.

Consider the effects of the distributed denial-of-service attack (DDOS) on the computer system of an air traffic control tower. Let us assume, that the attack may cause damage to the network components (e.g., unavailability of the system), cascade to damage to physical resources (e.g., crash of the airplane due to missing control data), and human injuries (e.g., result of the crash) [23]. To perform the Schmitt Analysis we need measurements of the factors, like severity. For example, we can state that the severity of the attack is "high" if its cascading effects cause injuries to more than 10 non-military, i.e., civilian, people. This can be represented with the following rule:

$$severity(a-id, targ, high) \leftarrow injured(x),\ x > 10,\ rank(civilian)$$

We may have several rules with the same head. Depending on the prerequisites of the rule (i.e., predicates of the body), more than one rule may be applicable. We briefly discuss how to handle logical inconsistencies (contradictory rules) and nondeterminism (ambiguity) in a subsequent section.

Damage Assessment First, we presents a way of calculating attack characteristics, using the concepts and their properties in the interoperable ontology and the formalization provided by ARP. We assume that each atomic resource (e.g., computer CPU unit) has a corresponding value and measure of atomic damage (i.e., the direct damage to this resource without the consideration of other

resources). Damage to a complex asset, say a personal computer, is calculated recursively, based on the damages of atomic resources. That is,

$$damage_c = f(f(damage_{c_1}), \ldots, f(damage_{c_l}))$$

where $c_1, \ldots, c_l$ are immediate subclasses or instances instances of the class c, and f is a function that defines how the damages to $c_1, \ldots, c_l$ contribute to the total damage. The recursion uses the ontology concept hierarchy to identify subcomponents of the assets.

Intuitively the calculation of the total damage of a system incorporates damages to the different components of the system. For example, a virus attack on the system may damage different software components, as well as cause damage because of the loss of production time, and the cost of recovery (software, human time, etc.)

Consequences of attacks rarely remain isolated. Cascading effects of cyber attacks must be addressed to gain a realistic evaluation of the damage caused by the attacks. See [14] for examples of cascading effects from kinetic and cyber attacks on a subway system. The total damage caused by a computer network attack, denoted as $DAMAGE_{attack}$, is calculated as

$$DAMAGE_{attack} = F[damage_{C_1}, damage_{C_2}, \ldots, damage_{C_j}]$$

where $damage_{C_1}$ is the damage on the original target computer system, $damage_{C_i}$ $(i = 2, \ldots, j)$ represent the damage to the indirectly effected assets. The function F determines how to combine the individual damages. As the simplest solution, both f and F may be defined as simple additions; that is, the total damage of an attack is the sum of the direct and indirect damages to the affected resources.

4.3 Conflict Resolution and Default Policy

Conflict resolution strategies are necessary if the overlapping concepts of the ontologies and the corresponding law create legal ambiguities (nondeterminism), illogical statements (inconsistencies), or no applicable law exists. A legal ambiguity may occur when more than one restrictions exist on the same concept. These restrictions are not conflicting but differ in other aspects (e.g., scope). In the logic framework this would result in nondeterministic behavior. Although the removal of such behavior is not the ultimate goal of our system, THEMIS will indicate such occurrences. This allows human experts to compensate for this ambiguity inside or outside the legal framework. A simple method to eliminate ambiguity is by assigning precedence criteria to the rules, which in turn allows to deterministically select the most prominent rules.

Inconsistencies occur when two contradictory restrictions can be derived over the same concept. Selection of the most prominent restriction may be similar to the ones mentioned for ambiguous restrictions. We are using legal hierarchy and rule-precedence settings to derive consistent decisions. Non-automated resolution of inconsistencies requires cooperation of legal expert and decision makers as well

as technologies to handle exceptions. Another conflict resolution strategy used by attorneys is called "most specific takes precedence." For instance, there is a general prohibition against shooting another citizen. This rule applies in almost all places at almost all times, but narrow exceptions do exist for self-defense and the defense of others. This structure ensures that *lacunae* in the law are kept to an absolute minimum.

A general principle is that there need not be any areas left uncovered by a rule of law. Most laws are not formulated to apply to narrow fact patterns, but rather to broad areas of similar circumstances. An entire field may be covered by a relatively simple rule, with narrowly defined subsets having specific rules to cover exceptional circumstances. THEMIS can be used to verify this desirable property. In encountering a situation that is not covered, THEMIS will generate and alert. An alert may indicate an incomplete policy specification in THEMIS or the non-existence of law with regard to this policy. Human analysis is recommended for further evaluation of the cause of an alert.

5 Conclusions

Computer attacks, due to their distributed nature, usually span multiple jurisdictions. Therefore, the evaluation of attacks, for the purpose of prosecution or response, requires the cooperation of each of the jurisdictions that the attack touches. THEMIS is an automated system whose purpose is to assist attorneys and those responsible for defending computing assets. THEMIS consists of legal ontologies which represent and retrieve domain knowledge about state, federal, and international law. We established the following requirements for THEMIS: it must address legal ambiguities with nondeterminism, it must provide a mechanism for identifying and resolving conflicts between laws, and it should allow for specifying default rules to capture the manner in which attorneys handle situations in which the law does not address one or more aspects of an attack. We intend to address each of the aforementioned requirements as we continue to refine our ontologies.

Acknowledgement

This work was partially supported by the National Science Foundation under grants IIS-0237782, CCR-0113515, the Department of Homeland Security through the Naval Postgraduate School Homeland Security Leadership Development Program, and the Critical Infrastructure Protection Project of the George Mason University.

References

1. Automated incident reporting (airCERT). Technical report, Carnegie Mellon Software Engineering Institute, Cert Coordination Center.

2. The rule markup initiative. http://www.ruleml.org/, 2004.
3. The Semantic Web. http://www.w3.org/2001/sw/, 2004.
4. Joint US/EU ad hoc Agent Markup Language Committee. SWRL: A semantic web rule language combining OWL and RuleML. http://www.daml.org/2003/11/swrl/, 2003.
5. P. Ammann, D. Wijesekera, and S. Kaushak. Scalable, graph based network vulnerability analysis. In *Proc. of the 9th ACM Conference on Computer and Communications Security*, pages 217–224, 2002.
6. K.D. Ashley. *Modeling Legal Argument: Reasoning with Cases and Hypotheticals.* Bradford Books/MIT Press, Cambridge, MA, 1990.
7. T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 2001.
8. K. Branting. Reasoning with portions of precedents. In *Proc. 3rd Intl. Conf. on Artificial Intelligence and Law*, pages 145–154. ACM Press, New York, 1991.
9. Harry Chen, Tim Finin, and Anupam Joshi. Using OWL in a pervasive computing broker. In *Workshop on Ontologies in Open Agent Systems, AAMAS.* citeseer.nj.nec.com/583175.html.
10. Benjamin N. Grosof. Representing e-business rules for the semantic web: Situated courteous logic programs in RuleML. In *Proc. Workshop on Information Technologies and Systems (WITS '01)*, 2001.
11. Auguston M. Rowe N. Michael, J. B. and R. D. Riehle. Software decoys: Intrusion detection and countermeasures. In *Proc. Workshop on Information Assurance*, pages 130–139. IEEE, 2002.
12. J. B. Michael. On the response policy of software decoys: Conducting software-based deception in the cyber battlespace. In *Proc. of the 26th Annual International Computer Software and Applications Conference*, pages 957–962. IEEE, 2002.
13. J. B. Michael, G. Fragkos, and M. Auguston. An experiment in software decoy design: Intrusion detection and countermeasures via system call instrumentation. In di Vimercati S. D. C. Samarati P. Gritzalis, D. and S. Katsikas, editors, *Security and Privacy in the Age of Uncertainty*, pages 253–264. Norwell, Mass.: Kluwer Academic Publishers, 2003.
14. J. B. Michael, G. Fragkos, and D. Wijesekera. Measured responses to cyber attacks using schmitt analysis: A case study of attack scenarios for a software-intensive system. In *Proc. Twenty-seventh Annual Int. Computer Software and Applications Conf.*, pages 621–627. IEEE, 2003.
15. J. B. Michael and T. C. Wingfield. Lawful cyber decoy policy. In di Vimercati S. D. C. Samarati P. Gritzalis, D. and S. Katsikas, editors, *Security and Privacy in the Age of Uncertainty*, pages 483–488. Norwell, Mass.: Kluwer Academic Publishers, 2003.
16. The Honeynet Project. *Know your Enemy - Revealing the Security Tools Tactic, and Motives of the Blackhat Community.* Addison-Wesley, 2002.
17. E. L. Rissland and K.D. Ashley. A case-based system for trade secrets law. In *Proc. 1st Intl. Conf. on Artificial Intelligence and Law*, pages 61–67. ACM Press, New York, 1987.
18. M.N. Schmitt. Computer network attack and the use of force in international law: Thoughts on a normative framework. Research Publication 1, Information Series, 1999.
19. D. B. Skalak and E. L. Rissland. Argument moves in a rule-guided domain. In *Proc. 3rd Intl. Conf. on Artificial Intelligence and Law*, pages 1–11. ACM Press, New York, 1991.

20. J. L. Undercoffer, A. Joshi, T. Finin, and John Pinkston. A target-centric ontology for intrusion detection: Using DAML+OIL to classify intrusive behaviors. *Knowledge Engineering Review – Special Issue on Ontologies for Distributed Systems*, 2004.
21. P. Visser and T. Bench-Capon. The formal specification of a legal ontology. In *Legal Knowledge Based Systems; foundations of legal knowledge systems. Proceedings JURIX'96. R.W.*, 1996. citeseer.ist.psu.edu/visser96formal.html.
22. P. Visser and T. Bench-Capon. A comparison of two legal ontologies. In *Working papers of the First International Workshop on Legal Ontologies.* University of Melbourne, Melbourne, Australia, 1997. citeseer.ist.psu.edu/visser97comparison.htm.
23. T. Wingfield. *The Law of Information Conflict: National Security Law in Cyberspace.* Aegis Research Corp., 2000.