Semantic Cyberthreat Modelling

Siri Bromander
mnemonic
Norway
siri@mnemonic.no

Audun Jøsang
University of Oslo
Norway
josang@ifi.uio.no

Martin Eian
mnemonic
Norway
meian@mnemonic.no

***Abstract*—Cybersecurity is a complex and dynamic area where multiple actors act against each other through computer networks largely without any commonly accepted rules of engagement. Well-managed cybersecurity operations need a clear terminology to describe threats, attacks and their origins. In addition, cybersecurity tools and technologies need semantic models to be able to automatically identify threats and to predict and detect attacks. This paper reviews terminology and models of cybersecurity operations, and proposes approaches for semantic modelling of cybersecurity threats and attacks.**

I. Introduction

When security incidents occur there is typically limited understanding of who the threat agent is, why they attack and how they operate, which makes it difficult to make well informed decisions about countermeasures. Threat agents who are not identified and made responsible for their actions will continue their criminal behaviour. When we do not understand the attacker we can only see - if even that- the results of the attacker's actions. Improved cybersecurity requires digital threat intelligence - structured and semi-automated analysis and sharing of information. In order to make sense out of increasingly large and complex datasets related to cybersecurity we see the potential in developing models and tools for automated or semi-automated classification and discovery of cyberthreats based on ontologies.

Semantic technologies and ontologies are a relatively new logic-based landscape of technologies and tools aimed at giving better meaning to large and unstructured corpuses of data. Interesting research challenges are for example to investigate semantic representations of relevant concepts in the domain of cybersecurity big data, in order to facilitate advanced machine learning, search and discovery.

The potential benefit of this approach is that the developed tools and related technologies will provide a flexible framework for representing and structuring the large variety of data with which security analysts are confronted. The framework can further be used for the implementation of cybersecurity analytics tools.

II. Cybersecurity Threat and Risk Models

Cybersecurity is the body of technologies, processes and practices designed to protect networks, computers, programs and data from attack, damage or unauthorized access. Cybersecurity thus assumes that some actors, typically called *threat agents*, have the intent and capacity to produce attacks, gain unauthorized access and cause damage. The magnitude of the perceived potential damage caused by cyber attacks is typically interpreted as security risk.

A. Specific Security Risk Model

Cybersecurity risks are caused by threats. However, the concept of a threat can be ambiguous in the sense that it can mean the threat agent itself, or it can mean the thing that a threat agent (potentially) produces, typically called a *threat scenario*. Figure 1 illustrates a specific risk model which integrates the concepts of threat agent and threat scenario.

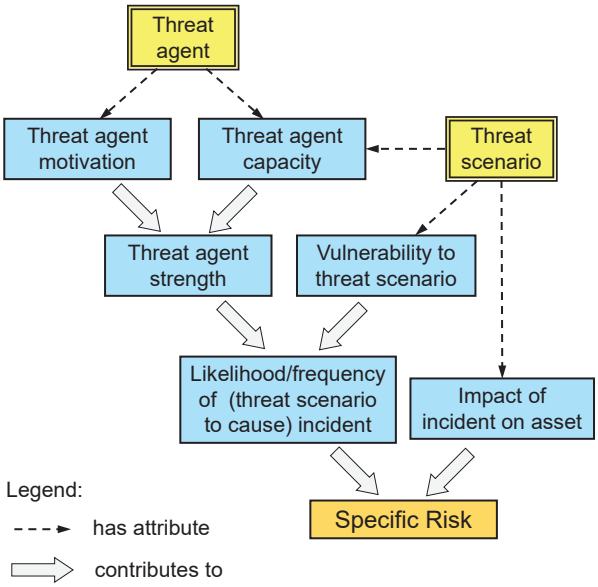

Figure 1. Specific risk model including threat agent and threat scenario

The specific risk model of Figure 1 emphasizes the risk dimension of threats, i.e. how threats lead to risk.

It can be seen that the threat agent and the threat scenario have very different attributes, but in combination they both contribute to risk. A threat agent can be modeled as a real agent with a motivation or goal as well as with a capacity to execute a specific threat scenario. Together, the motivation and capacity produce the strength of the threat agent. The threat agent strength can be modelled according to the weakest

This research was supported by the research projects TOCSA, ACT and Oslo Analytics funded by the Research Council of Norway.

link, i.e. the attacker is only as strong as the weakest of its motivation and capacity.

A threat scenario can be modelled as a sequence of attack steps which can be stopped by defence and security mechanisms. However, when the defence mechanisms fail to stop a specific threat scenario, we say that there are *vulnerabilities*.

The more severe the vulnerabilities and the greater the strength of the threat agent, the greater the likelihood that the threat scenario will cause a security incident and lead to damage, as illustrated in Figure 1. The actual risk of a specific threat scenario emerges by including the amplitude of the expected damage in case the security incident actually occurs. Risk assessment models such as in [1] are based on this interpretation of security risk.

There can of course be many different threat scenarios leading to the same goal when seen from the attacker's perspective. Each scenario represents the dynamic execution of a *tactic*. The attacker might consider multiple tactics, and then decide to use the one which is assumed to produce the greatest expected result with the least effort.

The threat scenario is an abstract set of steps executed in sequence, which from the victim/defender's perspective can cause damage to its assets. A threat scenario becomes a *cyber attack* when the scenario is actually executed. Behind every attack there is thus a specific threat scenario executed by an attacker or a group of attackers. However, a threat scenario by itself is abstract, and does not become an attack unless it is actually executed.

A threat scenario can therefore be interpreted as the blueprint for attacks. For cyber defenders there is thus a fundamental difference between detecting real attacks and identifying threat scenarios which only represent potential attacks.

B. Stillions' Detection Maturity Level Model

A model for the maturity of cyberthreat detection has been proposed by Ryan Stillions in several blogpostings [2]. A slightly extended version of Stillions' Detection Maturity Level (DML) model is illustrated in Figure 2. We have added the additional *DML-9 Attacker Identity* which can be important in certain contexts. We have also added *precision* and *robustness* to illustrate the qualitative aspects of features at each level. The DML model emphasizes the increasing level of abstraction in the detection of cyber attacks, where it is assumed that a security incident response team with low maturity and skills only will be able to detect attacks in terms of low level technical observations in a network, without necessarily understanding the significance of these observations. On the other hand, a security incident response team with high maturity and skills is assumed to be able to interpreted technical observations in networks in the sense that the type of attack, the attack methods used and possibly the identity of the attacker can be determined.

The levels of the DML model are briefly explained below. The focus is on what the IR team (incident response team) is

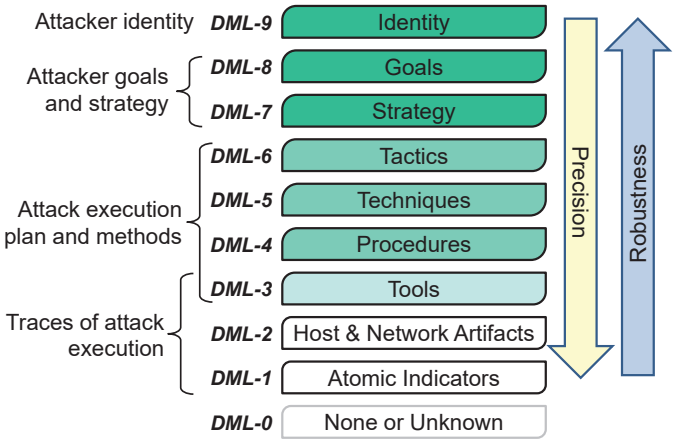

Figure 2. Detection Maturity Level Model [2]

capable of doing at each level. Our description is a summary and interpretation of Stillions' description [2].

- **DML-0 None or Unknown**. There is no IR team, or they are totally clueless.
- **DML-1 Atomic indicators of compromise (IOCs)**. These are elementary pieces of host & network artifacts, which might have been received from other parties. The value of atomic IOCs is limited due to the short 'shelf life' of this type of information.
- **DML-2 Host & Network Artifacts**. This is the type of information which can be collected by network and endpoint sensors. With high capacity links the amount of information collected can be overwhelming and requires good analytical tools to analyse and understand the attack at higher levels of abstraction.
- **DML-3 Tools**. Attackers install and use tools within the victim's network. The tools often change, so that a tool detected and analysed in a previous security incident might be similar but not exactly the same in new attacks. DML-3 means that the defender can reliably detect the attacker's tools, regardless of minor functionality changes to the tool, or differences in the artifacts and atomic indicators left behind by the tool.
- **DML-4 Procedures**. Detecting a procedure means detecting a sequence of two or more of the individual steps employed by the attacker. The goal here is to isolate activities that the attacker appears to perform methodically, two or more times during an incident. In the military jargon, procedures mean *"Standard, detailed steps that prescribe how to perform specific tasks"* [3].
- **DML-5 Techniques**. Techniques are specific ways of executing single steps of an attack. In the military jargon, techniques mean *"Non-prescriptive ways or methods used to perform missions, functions, or tasks"* [3].
- **DML-6 Tactics**. To detect a tactic means to understand how the attack has been designed and executed in terms the techniques, procedures and tools used. In the military jargon, tactics mean *"the employment and ordered*

arrangement of forces in relation to each other" [3].

- **DML-7 Strategy**. This is a non-technical high-level description of the planned attack. There are typically multiple different ways an attacker can achieve its goals, and the strategy defines which approach the threat agent should follow.
- **DML-8 Goals**. The motivation for the attack can be described as a goal. Depending on how the attacker is organised, the goal might not be known for the attack team executing the attack, the team might only receive a strategy to follow.
- **DML-9 Identity**. The identity of the attacker, or the threat agent, can be the name of a person, an organisation or a nation state. Sometimes, the identity can only be linked to other attacks without any other indication of who they are or from where they operate. The attacker identity might not be relevant to the defender if they only want to get the attacker out of the network. However, it is often important to be able to connect multiple attacks to the same actor in order to predict strategy, tactics, techniques and procedures expected to be used. This is an additional level defined by us, the original DML model [2] only consists of the levels 0–8.

The challenge is to leverage observed attack features detected at low levels to determine derivative causes at higher levels.

Assume that a given company *B* has as goal to beat company *A* in the open market. This goal might cause company *B* to use unethical means, with a strategy to steal secret information from company *A* in order to improve their own products and market position. Company *B*'s tactics may be to gain access to company *A*'s internal servers based on an attack plan with techniques, procedures and tools. Finally, the execution of the plan causes traces of the attack to be left in the network of victim *A*.

The cyber incident response team will first detect the traces, and from there must try to figure out what has happened and then decide the appropriate response. The traces are indicators, and the task of determining what really happened is a form of abductive reasoning which consists of using the indicators as classifiers to determine the nature and origin of the attack.

Most incident response teams of today are working on DML-1 and DML-2. Some are working on DML-3 and partly DML-6. However, the further up the stack you get the more seldom you find machine readable results from the analysis and work that is done. Defining semantic models for the type of information gathered in the higher levels of the DML model and the relations between them will enable more teams to increase their maturity level. Information sharing will also be facilitated by this development.

III. Elements of Semantic Threat Modelling

Discovering the real nature of a threat given a set of data or information requires a semantic model to represent all aspects of the threats with no room for ambiguous input. The further down the DML model you get, the more precise an identification can be done. The further up, the more costly a change is for the attacker and the more robust your conclusion of identity may become. Both aspects are useful for different roles and situations throughout a security incident. SIEM (Security Incident and Event Management) tools typically use semantic representation of host & network artifacts at the lower levels of the DML model, but rarely provide semantic representations of high level aspects. It is thus necessary to standardise the semantic representations of high level aspects in the DML model. This will allow automated reasoning to leverage the potential of machine learning and classifiers to do advanced cybersecurity analytical reasoning.

A. A Semantic Threat Classification Model

The primary focus of the DML model is to indicate levels of maturity in cyberthreat detection. However, the same model can be used as a basis for the design of cyberthreat classifiers, and we call this new model the *semantic threat classification model* (STCM).

Figure 3 shows the STCM which consists of a compact representation of the DML model combined with *classifiers* representing the analytical relationships from low level features to high level features.

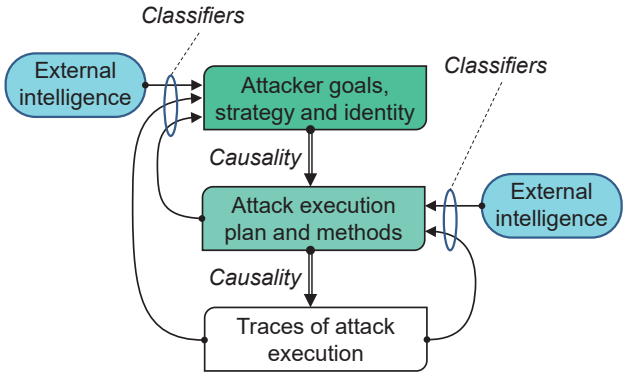

Figure 3. Semantic Threat Classification Model

Note that there are causal relationships from high level features to low level features. Hence, classifiers are used to reason in the opposite direction to that of causal relationships.

In machine learning and statistics, classifiers are used to determine categories to which some observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. For cybersecurity analytics, a classifier can e.g. be used to determine which type of attack a set of network artifacts belong to (i.e. are caused by), the goal of the attacker or even the identity of the attacker.

Note that contextual information can also be used as input indicators for classifiers. Contextual intelligence can e.g. be political events covered by the media. A political conflict between nation states can make it more likely that states launch specific types of cyberattacks against each other.

The challenge for developing reliable classifiers is to identify appropriate semantic features and their variables at each

level of abstraction, and to have available sufficient amount and type of data in order to give the classifiers sufficient training for reliable detection and classification.

The design of classifiers for machine learning is heavily dependent on statistical methods, and several authors have pointed out the importance of mathematics for cybersecurity [4].

B. Semantic Feature Extraction

Stillions' DML model [2] uses English prose to informally define each level of abstraction. The use of classifiers, however, requires formal definitions of the features at each level of abstraction. Our approach is to gather informal descriptions of goals, strategies, tactics, techniques and procedures from the literature. Through analysis of these informal descriptions, we derive tuples that describe each level of abstraction. In the following, we illustrate this process for the abstraction level "Goals".

Stillions mentions the following goal as an example:

> Replicate Acme Company's Super Awesome Product Foo in 2 years or less [2]

If we ignore the time dimension of this goal, then we can derive the 2-tuple ("Replicate", "Product") from the informal description.

From Mandiant's APT1 report [5], we can derive the following goals: ("Replicate", "Product"), ("Replicate", "Manufacturing process"), ("Obtain", "Business plan"), ("Obtain", "Policy position").

Another goal can be derived from Symantec's blog post on the "Cadelle" and "Chafer" APT groups [6]: ("Monitor", "Individuals").

By generalising the examples above, we get the following definition of a goal: (Action, Object). When we observe the 2-tuples from the examples, we identify two challenges. The first challenge is that we use strings to describe each element of the tuple. If we use 2-tuples of strings in a system where a multitude of analysts and classifiers identify and record new goals, then the result will be duplicated by synonyms resulting in an explosion of features. In order to avoid this, our goal is to define a formal taxonomy of goals, where each tuple contains references to the taxonomy.

The second challenge is that the second element of the 2-tuple is too general. To alleviate this, we must define sub-elements that are more specific, e.g. that the "Product" in the first example is manufactured by "Acme company", and that the specific product is "Super Awesome Product Foo". In the last example, "Individuals" could have a sub-element "Iranian Citizens". Note that in some cases we will not be able to determine these sub-elements due to insufficient data.

Applying this approach to all the layers of abstraction in the extended DML model requires a monumental amount of effort. We believe that in order to achieve this, a community effort is needed. Thus, one of our primary goals is to lay the foundations for such an effort. Furthermore, re-using existing standards and taxonomies where applicable can significantly reduce the amount of work needed. A good example of such re-use can be observed for the abstraction level "Techniques". The MITRE ATT&CK taxonomy [7] has already defined more than 100 techniques used by adversaries in the post-compromise phases of an attack.

C. Current Initiatives for Cyberthreat Representation

There are several initiatives currently being used for representation and sharing of data on the different levels of the DML model. The following initiatives are seen as useful and may be used when selecting features for representation on the different levels:

- **INTEL Threat Agent Library** (TAL) [8] was suggested in 2007 and provides a consistent reference describing the human agents that pose threats to IT systems and other information assets. This library may serve as a feature of "Identity" in our semantic threat modelling.
- **STIX** [9] is a language for having a standardized communication for the representation of cyberthreat information. It is well known in the incident response community, but not serving the purpose of describing all aspects of cyber threats. The main shortcoming in the current version is the lack of separation between tactics, techniques and procedures.
- **CAPEC** The objective of the Common Attack Pattern Enumeration and Classification (CAPEC) [10] effort is to provide a publicly available catalog of common attack patterns classified in an intuitive manner, along with a comprehensive schema for describing related attacks and sharing information about them. CAPEC is run by MITRE and is openly available for use and development for the public. For our semantic threat modelling it may be used when describing 'Tactics' and 'Techniques'.
- **ATT&CK** is a common reference for post-compromise tactics, techniques and tools [7] run by MITRE. ATT&CK and CAPEC are related and do not exclude use of each other.

IV. Example applications of semantic cyberthreat models

In this paper, we argue that semantic cyberthreat models can help cybersecurity professionals to be more effective and efficient. This section presents some concrete examples from our own experience that support this hypothesis.

A. Incident response

Breaches due to attacks from advanced persistent threats (APTs) are often detected post-compromise. APTs quickly initiate lateral movement after the initial compromise, so assessing the scope of the breach can be challenging. In order to assess the scope of the breach, we need to know how the threat agent operates and what kind of indicators, artifacts, tools, tactics, techniques and procedures (TTPs) we should search for. The incident response analysis process typically consists of the following steps:

1) Evidence collection
2) Analysis of evidence

3) Identification of new indicators, artifacts, tools and TTPs
4) Threat agent attribution

Steps 1-3 are performed in an iterative fashion. The analysis results may indicate that we need to collect more evidence, or that we should search the existing evidence for new indicators. If we are able to perform step 4 and attribute the breach to a known threat agent, then we can leverage our historical knowledge of this threat agent. We can use this knowledge to guide our evidence collection and analysis. We have used the MITRE ATT&CK taxonomy [7] to be able to quickly compare our evidence to known threat agents during incident response. By manual analysis, we found threat agents that used tools and techniques very similar to what we observed in our evidence. The ATT&CK taxonomy [7] has a loose semantic model connecting threat agents, tactics, techniques and tools. It does not model procedures, artifacts or indicators. In order to automate the analysis of threat agent similarities, we implemented a simple semantic model using a graph database. The model linked threat agents to observed indicators, artifacts, tools and TTPs. We then used the graph database to find all subgraphs that connected the findings from our incident to known threat agents. The result enabled us to attribute the evidence from our incident to a known threat agent, and the results helped guide our evidence collection and analysis. Another great advantage of using such a model is that the attribution hypothesis can be re-tested as more knowledge is added to the graph, in order to avoid confirmation bias. Our experience from this incident was that we were able to attribute the evidence to a known threat agent much more rapidly than by using manual analysis. We were also able to fully document all relations between our evidence and the threat agent by issuing a simple graph query.

B. Requests for information

A common task for threat intelligence analysts is to find all information related to a single data point, e.g. an IP address, a malware sample or a threat agent. Having a semantic model implemented as a graph makes it possible to complete such a task quickly and reliably by issuing a single graph query.

C. Intrusion detection

Current intrusion detection systems operate at DML-1, DML-2 and/or DML-3. One of the challenges with operating at DML-4 and above is that TTPs are commonly described using English prose, i.e. as unstructured data. This makes it challenging to translate the description to intrusion detection signatures, and signature development must be performed manually. Defining formal models for TTPs makes it possible to automatically generate signatures from structured data when a new TTP is defined. One concrete example is the procedure described in [11]:

> An example would be an adversary running **net time**, followed by the **AT.exe** command to schedule a job to kick off just one minute after the current local time of the victim system. [11]

Given an endpoint security solution that logs process execution with arguments and command inputs/outputs, a human analyst could write a signature to detect this procedure. The signature would have to detect the following:

1) Execution of **net.exe** with **time** as the first argument and **victim system** as the second argument
2) Timestamp returned by the command in step 1
3) Execution of **at.exe** with **victim system** as the first argument and ((timestamp from step 2) + 1 minute) as the second argument

Interpreting the description "to schedule a job to kick off just one minute after the current local time of the victim system" is easy for a human, but very difficult for a computer. A formal definition of this procedure would make it possible for a computer to automatically generate signatures for the procedure by applying transformation rules.

V. Conclusion

Semantic modelling of threats is a promising approach for automated threat and attack detection at multiple levels of abstraction. A semantic model of threats will enable security analysts to work faster and more efficiently in terms of identifying threat agents and take advantage of previous experience and gathered intelligence when handling incidents caused by known or unknown threat agents. The task of extracting semantic features for all levels of abstraction in our suggested extended DML model is an undertaking of daunting proportions. In order to make this task manageable the reuse of related standards and taxonomies is required.

References

[1] ISO, *ISO/IEC 27005:2011 - Information technology – Security Techniques – Information security risk management (second edition)*. ISO/IEC, 2011.

[2] R. Stillions, "The DML Model," http://ryanstillions.blogspot.com/2014/04/the-dml-model_21.html, 22 April 2014.

[3] U. DoD, *Department of Defense Dictionary of Military and Associated Terms*. Joint Chiefs of Staff, 2010.

[4] A. Pinto, "Secure because of Math: A deep-dive on Machine Learning-Based Monitoring," Black Hat Briefing. BlackHat Conference, 2014.

[5] Mandiant, "Mandiant APT1," https://www.fireeye.com/content/dam/fireeye-www/services/pdfs/mandiant-apt1-report.pdf, 18 February 2013.

[6] S. S. Response, "Iran-based attackers use back door threats to spy on Middle Eastern targets," http://www.symantec.com/connect/blogs/iran-based-attackers-use-back-door-threats-spy-middle-eastern-targets, 7 December 2015.

[7] MITRE, "Adversarial Tactics, Techniques and Common Knowledge (ATT&CK)," https://attack.mitre.org/.

[8] T. Casey, "Threat agent library helps identify information security risks," *Intel White Paper, September*, 2007.

[9] S. Barnum, "Standardizing cyber threat intelligence information with the Structured Threat Information eXpression (STIX)," *MITRE Corporation*, vol. 11, 2012.

[10] MITRE, "Common Attack Pattern Enumeration and Classification (CAPEC)," https://capec.mitre.org/.

[11] R. Stillions, "On TTPs," http://ryanstillions.blogspot.com/2014/04/on-ttps.html, 22 April 2014.