On properties of cyberattacks and their nuances

Jassim Happa and Michael Goldsmith
Department of Computer Science, University of Oxford, Oxford, UK

Received 21 April 2017
Revised 8 May 2017
Accepted 10 May 2017

Abstract

Purpose – Several attack models attempt to describe behaviours of attacks with the intent to understand and combat them better. However, all models are to some degree incomplete. They may lack insight about minor variations about attacks that are observed in the real world (but are not described in the model). This may lead to similar attacks being classified as the same type of attack, or in some cases the same instance of attack. The appropriate solution would be to modify the model or replace it entirely. However, doing so may be undesirable as the model may work well for most cases or time and resource constraints may factor in as well. This paper aims to explore the potential value of adding information about attacks and attackers to existing models.

Design/methodology/approach – This paper investigates used cases of minor variations in attacks and how it may and may not be appropriate to communicate subtle differences in existing attack models through the use of annotations. In particular, the authors investigate commonalities across a range of existing models and identify where and how annotations may be helpful.

Findings – The authors propose that nuances (of attack properties) can be appended as annotations to existing attack models. Using annotations appropriately should enable analysts and researchers to express subtle but important variations in attacks that may not fit the model currently being used.

Research limitations/implications – This work only demonstrated a few simple, generic examples. In the future, the authors intend to investigate how this annotation approach can be extended further. Particularly, they intend to explore how annotations can be created computationally; the authors wish to obtain feedback from security analysts through interviews, identify where potential biases may arise and identify other real-world applications.

Originality/value – The value of this paper is that the authors demonstrate how annotations may help analysts communicate and ask better questions during identification of unknown aspects of attacks faster, e.g. as a means of storing mental notes in a structured manner, especially while facing zero-day attacks when information is incomplete.

Keywords Attack models, Cyberattacks

Paper type Conceptual paper

Introduction

The evolution of digital systems has made it difficult to discuss cyberattacks across a variety of stakeholders. People of different backgrounds, including politicians, ethicists, lawyers, business owners and other stakeholders may all be decision makers in the face of cyberattacks. There is currently no universal language or model that is suitably able to cater for the needs of all stakeholders who may be affected by an attack. This means that

PSU Research Review
Vol. 1 No. 2, 2017
pp. 76-90
Emerald Publishing Limited
2399-1747
DOI 10.1108/PRR-04-2017-0024

stakeholders have to be technical and comprehend many other non-technical aspects of attacks or else misunderstandings across the stakeholders are likely to happen. Attack models tend to describe attacks at a technical level, whereas it is becoming increasingly important to consider the real-world, non-technical effects of cyberattacks as well. These may not be straightforward to identify from attack models presently. Happa and Fairclough (2017) discussed this issue in-depth and proposed that decision makers should compensate for their knowledge limitations by building a common foundation to understand attacks across a variety of stakeholders through the use of mental models (i.e. place more emphasis on insight that is outside one's own expertise). The paper proposed a first version of a framework that enables more meaningful, in-depth discussions about cyberattacks, irrespective of stakeholder background. The authors also outlined possible extensions. One such extension was to look at attack modelling from a pragmatic point of view when time and other resources are extremely limited (Happa *et al.*, 2016).

Cyberattacks are becoming increasingly complex by the year. In the past few decades, the changes in the technology landscape has led to a growth in types of attacks, beyond the technological layer, such as advanced persistent threats, social engineering attacks, insider threats as well as the more traditional attacks on network assets. To understand and combat modern attacks, it is essential to consider a variety of aspects of attacks. Many attack models and frameworks exist in the effort to describe behaviours of attacks to understand and (thus) combating them better. Several challenges emerge when using these models however. All models are to some degree incomplete and require continual updating to describe details about attacks so they do not become outdated. Moreover, they often lack nuanced insights about attack parameters, environments or other variables. Lack of such information may reveal that two instances of an attack could be classified as the same one when in some regard they are not. It may also lead to misunderstandings about what makes up an attack when communicating the model to a wider stakeholder audience. The appropriate response would be to replace or modify the model to accommodate for corner cases. However, changing an established model may be undesirable, as the model may work well for most cases of attacks, or time and resource constraints may factor in as well when deciding to continue using a model or moving on to another.

Motivation

The purpose of this conceptual paper is to investigate how analysts and academics can better understand and use subtle differences in attack parameters. We do so by first identifying commonalities across a broad range of existing attack models. From this, we propose a novel approach to storing mental notes in the form as annotations (akin to Nota Bene): a means to resolve a large part of incomplete-modelling issues at least temporarily for an analyst or researchers using an attack model.

This paper explores the uses of annotations to describe nuances of cyberattacks and how it may (as well as how and when it may not) be an appropriate solution to communicate subtle differences between attacks. However, to be clear: we do believe that annotations are something that are already used substantially by analysts and researchers (albeit informally). This paper summarises the various uses of annotation in attack models and explores how readers can best leverage them, including making use of a novel annotation table.

The contribution of this paper is that we propose an approach that may help analysts communicate and ask better questions during identification of unknown aspects of attacks faster through annotations, for example, as a means of storing mental notes in a structured manner, especially while facing zero-day attacks when information is incomplete.

The paper is structured as follows. Section 2 presents related work, focusing on existing attack models and commonalities across them. Section 3 details properties, nuances and annotations in attack models and presents an in-depth discussion on the matter. Finally, Section 4 concludes the paper and outlines future work.

Related work

This section overviews existing attack models. We consider those relating to computer networks and those that may involve people as part of the attack and have therefore chosen to not cover more specialised attack models (e.g. specific to wireless sensor networks or cyber-physical systems only).

Early works

Models, taxonomies and descriptors can be used to examine attacks. Early works in the topic of attack models focus on documenting and understanding vulnerabilities of systems, such as the *Research in Secured Operating Systems* (Abbott *et al.*, 1976) and *Protection Analysis* (PA) (Bisbey and Hollingsworth, 1978). The goal of the PA project was to collect examples from abstract patterns. According to their report, more than 100 examples were recorded. The errors that were found in the systems were classified into four main categories: *domain errors* (e.g. exposed or incomplete destruction of data or context of data), *validation errors (e*.g. boundary condition errors and failure to validate), *naming errors* (e.g. aliasing and incomplete revocation of access) and *serialisation errors* (e.g. reference errors and interrupted atomic operations).

Abbott *et al.* (1976) classified vulnerabilities in terms of *incomplete parameter validation, inconsistent parameter validation, implicit sharing of confidential data, inadequate serialisation/asynchronous validation, inadequate authentication, violable prohibition* (e.g. manipulating data outside one's domain) and *exploitable logic error* (e.g. errors in routines that end up giving privileged access). Bishop and Bailey (1996) did a critical analysis of the two stating both also suffer from some ambiguity as some vulnerabilities may fall within the different classes.

Bishop (1995) proposed a vulnerability taxonomy of UNIX systems. It was classified into six different vulnerability axes, including:

(1) *Nature* – the type of flaw is described using the aforementioned PA categories.

(2) *Time of introduction* – when a vulnerability was added or known.

(3) *Exploitation domain* – what is gained through an exploitation.

(4) *Effect domain* – what can be affected by a vulnerability.

(5) *Minimum number* – the minimum number of components necessary to exploit a vulnerability.

(6) *Source* – the source of identification of a vulnerability.

Cohen (1997) described nearly 100 different classes of attacks gathered from many examples. Cohen follows on to outline what each of these separate attacks may involve as well as the defensive means to combat attacks. His work on classifying attacks discussed the idea of attacks having different *properties*, such as:

- *Non-orthogonality*: Malware can be many things, for example, a virus can be a worm, but it can also be a trojan.

- *Synergy*: When combining attacks, standard statistical techniques may not be effective to analyse them, as two attacks combined may yield a worse (or better) net effectiveness than if the attacks were executed separately.
- *Non-specificity*: Attacks are often hardware independent and rely on the implementation of protocols to be executed.
- *Descriptive only*: It is difficult to provide a detailed formal description about an attack.
- *Limited applicability*: Historical profiles of attacks may lead us to believe that certain attacks or items (about attacks) are more or less important than others. However, there is no universally accepted method that allow us to straightforwardly predict the circumstances under which an attack will be successful. This is still a subjective assessment. It should be mentioned however that despite this general property, in many cases, in-depth analysis (e.g. malware source-code) should produce reasonable predictions fast.
- *Incompleteness*: Inability to get a comprehensive understanding about an attack immediately and need to characterise behaviours in mathematical form to obtain a complete understanding of an attack.

Development of standards

In a continually evolving threat landscape, the detection and prevention of attacks becomes an increasingly complex task. Cohen's work highlights how considering properties of attacks can be useful in the interest of attempting to be comprehensive in our understanding about the intrinsic characteristics about attacks and defences. While only a few academic works have continued investigating attack properties, substantial progress has been made in the effort to attempt to make security more measurable with standards such as:

- FIRST's Common Vulnerability Scoring System (Grance, 2006) addresses concepts such as attack complexity, scope, user interaction, assumptions about privileges required, but also introduces temporal scores such as exploit code maturity, remediation levels and report confidence. Finally, there is also an environmental score that considers modified requirements based on the environment the vulnerability is executed in.
- Mitre's OVAL efforts such as the Common Vulnerability Enumeration (CVE) (Mann, 1999), Common Weakness Enumeration (CWE) (Martin, 2007), Common Attack Pattern Enumeration and Classification (CAPEC) (Barnum, 2008) and Malware Attribute Enumeration and Characterization (Obrst *et al.*, 2012) are some examples of Mitre's efforts to describe attacks.

Informal descriptions

The CIA-triad: *Confidentiality, Integrity and Availability* (CIA) NIST (2004) is often used to describe how an attack has compromised a system. NIST's article discusses the CIA triad in terms of severity of impact (high, medium, low). The CIA triad is widely used but a rudimentary means to broadly categorise the net effect of an attack. It can in many ways be considered to be simplistic. For instance, the fact that availability has been compromised in a system, provides no description about *how* exactly availability has been affected.

Availability could be stopped temporarily, slowed down or stopped completely. Such details about the descriptors are generally not covered by CIA triad.

Formal methods

Security protocol verification is also a means of understanding attackers using process algebra (such as Communicating Sequential Processes) and model checking. More formal approaches problem enable analysts and researchers to identifying how, for instance, that despite good cryptographic algorithms being important, poor protocol design can still render a software application open to exploitation (Ryan *et al.*, 2000; Roscoe and Goldsmith, 1997), with the common example used being a man-in-the-middle attacks. Quantifying attack actions is a key work in cyberattack analysis. A number of mathematical models aim to describe behaviour of cyberattacks which aim to compromise the CIA of a system, including ability to compute probabilities of security failure due to violations of different security attributes (Madan *et al.*, 2004), stochastic modelling using intrusion processes (Almasizadeh and Azgomi, 2009), DoS attack scheduling with energy constraints (Zhang *et al.*, 2015), false data injection attacks (Deng *et al.*, 2017).

Hierarchy structures and linear processes

Many attack models currently describe attacks either as linear processes or as hierarchical structures. Linear processes assume that one action happens after another and thus capture the temporal element of attacks. Three examples are the Howard and Longstaff (1998), Hutchins *et al.* (2011) and Happa *et al.* (2016) models. Howard and Longstaff, for instance, grouped properties in relation to an attack's motivation and objectives. Five stages are considered: attackers (who – threat actors), tools (what – implementation or design), access (how – actions and targets), results (outcome – technical) and objectives (why – purpose of the attack).

Hutchins *et al.* published the cyber kill chain a model that identifies what adversaries must complete to achieve their objective. The model describes a seven-step process that the attacker needs to execute, including:

- *Reconnaissance* – harvesting information about a system to attack it;
- *Weaponisation* – concocting an exploit based on the prior reconnaissance work that is likely to be successful;
- *Delivery* – the actual sending of the weaponised exploit;
- *Exploitation* – the successful break-in;
- *Installation* – the successful installation of malware on an asset;
- *Command and control* – the setting up a remote channel so an attacker can execute an attack from the outside; and
- *Act on objectives* – when the attacker actually executes their intended attack.

Happa *et al.* (2016) proposed an event-driven response model which aims to help analysts identify how to best respond by traversing the attack backwards (on a per-severe event basis) impact (*what is attacked?*), vector (*how is it attacked?*), motives (*why is it attacked?*) and attribution (*who has attacked?*) and proposes that analysts can use ad-hoc methods (any model or method available to them) to reverse-engineer details where and if they exist.

Linear processes describe easy-to-follow procedures that attackers may take, they often fail to describe lateral movement, concurrent attacks or attack steps and, more importantly, how those relate to each other. Attack graphs have been considered a means to address this

issue. Attack graphs facilitate quantitative study of attackers. Attack graphs can act as a map of nodes on a network, with each node representing a point of entry or vulnerability, and showing multiple steps that an attacker can take to achieve their goals. Several approaches to attack graphs exist. Wang *et al.* (2006) demonstrate how poor interpretation of security metrics can decrease security through attack graphs. Homer *et al.* (2009) presented an attack graph using probabilistic approach to quantify risk. Janse van Rensburg *et al.* (2016) use attack graphs to consider the security of a network from the perspective of many attackers simultaneously using deterministic and probabilistic measures.

Hierarchical structures, on the other hand, describe attacks in terms of many different constituent components of the attack and attempts to build a taxonomy that describes those properties relate to each other. They have the advantage of describing attacks in terms of their different properties but often neglect the temporal component. These structures may not necessarily describe how their respective building blocks work together to form an attack.

Examples include the following:

- *Landwehr* (Landwehr *et al.*, 1994) follows early works related to vulnerability classifications and builds hierarchies of these vulnerabilities in a manner akin to a treemap.
- *AVOIDIT* (Simmons *et al.*, 2009) is a building-block taxonomy that make up attacks related to the attack vector, operational impact, defences, informational impact and targets.
- *CAPEC* (Barnum, 2008) classifies attacks based on shared characteristics in a hierarchical approach. For example, an SQL-injection attack is a type of code-injection attack. Attack patterns are modelled after object-oriented design patterns and exclude low-level implementation details by design.
- *VERIS* (VERIS, 2015) is a schema that aims to provide a framework for a consistent approach to documenting and considering attacks enumerations at different levels, including information about actors, actions, attributes, assets among other things.

Domain-specific models and frameworks

While these efforts show substantial progress in tackling cyberattacks and understanding, in-depth, about attack steps and behaviours, they may not be feasible for all circumstances, especially when new attack vectors are introduced and, more importantly, require continual renewing. In the most recent years, the literature has seen more specialised, domain-specific attack models, particularly popular in more recent years is the insider attack domain, looking specifically at the human element of cyberattacks (Magklaras and Furnell, 2005; Kandias *et al.*, 2010; Legg *et al.*, 2013). Legg *et al.* (2013), for instance, outlined a taxonomy describing how insider threats may conduct their attacks. The model describes the domain in which an insider can operate into four key areas: the *enterprise* level (e.g. the mission of the organisation), the *people* level (e.g. the psychology and human factors), *the information and technology* level (e.g. the software and data present) and, finally, the *physical* level (e.g. the hardware and buildings). The model outlines how insiders can exploit all areas and how the intersection across all domains can be considered in the attempt to identify attacks. Agrafiotis *et al.* (2015) extended this work by considering how attack graphs can be used to express steps that (historically speaking) other insider attackers have taken and how this information can be used again in the future to identify new attacks by insider threats.

Using nuances and annotations in attack models

This work looks into the practical components of attack model interpretation – an often overlooked component in attack models. In this particular paper, we discuss considerations and implications of uses of annotations to described nuances in attacks. As outlined in the related work section: attack models often summarise goals (what), means (how), motives (why) and some assumptions about attribution (who), and, while many models and frameworks currently are suitably capable of expressing rich details about attacks for many circumstances, all are to some degree incomplete. From the models we have reviewed so far, it is possible to identify a number of noteworthy commonalities and differences.

When using any model to describe an attack, we foresee a feedback loop akin to the one shown in Figure 1. The analyst begins by investigating data until an attack has been detected. Then speculate details about the attack, including the incident input and output parameters and environment, identify them one by one until all that matches the model (including mental model) have been found. The results can then be shared with other analysts, then verified and refined.

Attack model generally details specific activities leading to the compromising of an asset. Models often focus on describing how to overcome a barrier of sorts, whether this be the vulnerability that can be exploited or describing a method of approach to exhaust or flood a system (e.g. attacking a system's weakness – an intrinsic concern with the system that cannot be patched). However, exactly what this barrier is, is specific to each attack model or framework. While some [e.g. the *Killchain* (Hutchins *et al.*, 2011)] place more emphasis on the types of steps and characterising what goal each step is seeking to reach, others (e.g. VERIS, 2015) adopt a more metrics-based approach to provide a language for describing security incidents in a structured and repeatable manner. Moreover, many

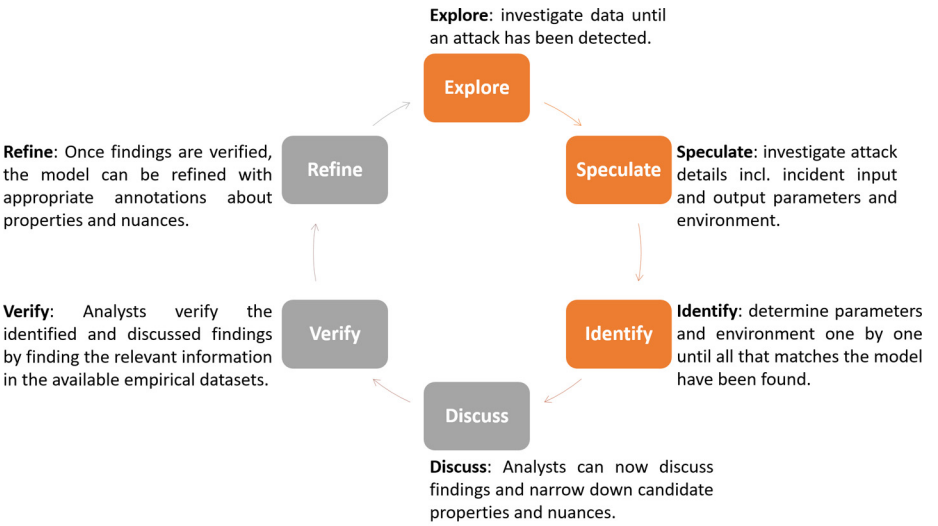

Figure 1.
The feedback look of using an attack model to understand an attack

Notes: It works for either when a model describes a particular use case or when describing more generic attacks; Orange boxes indicate steps that involve work done by one single analysts (either in parallel or independently); Grey boxes indicate (potentially) collaborative efforts

models may not be able to distinguish between subtle differences in attack input parameters, attacker/defender restrictions, the environment or other factors.

Challenges using attack models currently

Existing research has several key issues that needs to be addressed. Currently, it is not straightforward to:

- *include nuanced information in models.* This issue is often a modelling problem: abstraction provides inclusivity as it can consider more attacks in any description, but sacrifices detail, which results in potential lack of precision and accuracy.
- *include information that is difficult to quantify in models.* As Cohen's work suggests, some attacks have the property of being "descriptive only", and it is difficult to provide a detailed formal description about the attack. This may be partly due to what Howard & Longstaff suggested in that there is no common language for attack incidents. With each model using their own definitions, the cybersecurity language space needs time to mature in order for people to understand how we can go from qualitative descriptions to quantitative (ideally measurable) ones.
- *ensure a model is able to remain valid despite changes in the threat landscape.* Many models also require continual updating otherwise they become outdated, this especially applies to hierarchical structures that have not accounted for types of attacks that do not exist during the writing of the article.
- *compare insight or notes across from different models.* Few attack models, frameworks or methodologies refer to the same exactly same definitions. To compare notes across different models, it is necessary to keep a record (mental note, reference book or otherwise) to allow for insight to be used across models.
- *include issues related to known unknowns.* If, for instance, an aspect of an attack is not fully understood, but some observations about that aspect has been made, the analyst or academic should be able to append this information to an instance of a model under consideration, enabling them to at least recognise where certain insights about aspects of an attack needs to be improved.
- *creatively use attack models as an exploratory approach.* Research involves some level of creativity. Finding new ways to detect and describe attacks involves research, some creativity is therefore necessary as part of the thinking process. Attack models currently assume that academic or scientific rigour is crucial for the correct interpretation of an attack. We argue that to reach this point, it can be facilitated by a phase in which creative processes (bound by past experiences in attack models) can facilitate cognitive processes related to considering how attackers may behave for any particular instance.

Howard and Longstaff's (1998) taxonomy is able to specify incident parameters in a relatively flexible manner, making it adept to change with the threat landscape. For a particular instance of an incident, it not only is able to capture several of the actions within an incident but also sheds light on the reason for an attack (e.g. for financial gain, to cause system damage or for political gain), and analysts are able to inject their own thinking into a model. The model breaks down attacks into attacks and events. Events are a sub-system of

an attack that requires an action and a target, however, the other items in the attack simply describe the inputs and outputs of the aforementioned event.

The model can describe incidents in terms of matrices or almost like a sentence in its structure. The order "Attackers, Tool, Vulnerability, Action, Target, Unauthorised Result, Objectives" enables academics or analysts to specify the details of the attacks themselves. The examples provided remain relatively high-level, but the end results are a sentence structure that could be used to outline key stages in an attack. This almost forms a high-level sentence. One example incident might be: "*Hackers, Toolkit, Implementation vulnerability, Steal, Data, Disclosure of Information, Damage*" – already provide a lot of information about an attack.

Which can be re-phrased as:

> Hackers from <groupname> used the <name> toolkit to exploit an implementation vulnerability in <software name> to steal sensitive data from <organisation name> servers to gain disclosure of information they should not be privy to that leaves <organisation> with significant reputational damage.

Even with this insight, the information provided can still be too high level to be actionable. The model is explicit in terms of who attackers and their objectives might be (a full list is provided in their publication), and allow for detailed classification terms such as reporting/starting/ending dates, details of locations, etc. to be added. However, the model considers incidents in isolation, whereas in reality there may be concurrent incidents or collusion in effect (several attacks at the same time). Cohen, on the other hand, drops the notion of stages states and simply investigates the notions of intrinsic characteristics about the attacks as properties that may or may not exist. This is useful but does not necessarily provide a view of the attack narrative (i.e. the stages involved).

Properties and nuances

We propose that properties of attacks can have nuances. A nuance here is not intended to be a new concept in cybersecurity, but, rather, it is a word of convenience used in this paper simply to denote the ranges of values that a property (or set of properties) may have in an instance of an attack (not specifically limited to Cohen's idea of Properties either, but we did take Cohen's work as a starting point). More generally speaking, nuances relate to inputs, outputs, restrictions, assumptions or environments of an attack. Identifying a comprehensive list of nuances would be an exhaustive task. This paper will instead provide a set of tangible examples of nuances and propose it is up to each analyst or academic to identify other nuances and annotations of interest should they wish to adopt this approach – indeed, which properties and nuances truly matter in the context of attack models is subject to future work. For now, nuances should be considered as thought-experiments, facilitating the discussion and understanding of the attacks through the attack model and used on an ad hoc basis. Examples of nuances might be asking detailed questions in a model about:

- *Measurability of an attack aspect* – Are we able to measure (for instance) impact, vector, motive and attribution of an attack: listing people, physical items, hardware, software, finance, reputation, etc. (more importantly, how those relate to properties)? Are we able to measure the vectors the attack takes and how much of an overhead the attack has on the vector itself? Can we measure motives or any information about attribution from any aspect of the attack?
- *Influence of an attack aspect* – What indirect and direct sway on the system does the property or aspect of the attack have?

- *Duration of the attack aspect* – Is the nuance limited for a time period, if so how long, is it recurring? How long does the impact last? How long does it take to execute the attack from the attacker's or defender's perspective? Is the longevity of the impact affected by environment factors (e.g. how fast can an organisation recover if they have the right measures in place?)
- *Transparency of the attack aspect* – How visible is the aspect to the defender and the attacker?
- *Repeatability* – How repeatable is the attack, and can we measure the performance of the attack – if in a controlled environment, are we able to recreate the attack and learn more about the attacks nuances in terms of which conditions have to be satisfied for the attack to be repeatable?

We consider an *aspect* of any attack here to be any phase of an attack. In the case of kill chain, an aspect could be *reconnaissance*, in the case of Happa *et al.'s* (2016) model it could be *impact*, *vector*, *motives* or *attribution*, any stage in Howard & Longstaff's model or any key category in Mitre's efforts, etc.

Nuances derived from speculation and evidence

Nuances can be used to describe the ranges of values for each of the aspects. We believe it is up to the analyst on how exactly annotations are used to describe nuances, allowing for flexibility. Annotations could be either:

- *highly speculative* – they are informal thoughts-in-progress, subjective or qualitative assessments;
- *with evidence* – there has to be evidence to support the comment, either an observation or repeated; and
- *a combination* of being both speculative and with evidence.

In the case of Cohen's properties, we may list the various non-orthogonal ways a piece of malware can be classified as (e.g. a virus, a worm, a trojan, etc.). Similarly, the synergy property of an attack, the ranges of synergy enable us to express the nuances that have been observed between two or more attack vectors. The means to annotate could take the form of a mind map or a list of items describing how the particular attack relates to each property. With Howard & Longstaff's (1998) model, it would be possible to append insight about the incident that the model has not taken into account for. In a Howard & Longstaff example, assuming there exists contextual relevant to the attack, it should be considered. For instance, when considering the keywords (of a particular attack): Hackers[1], Toolkit[2], Implementation vulnerability, Steal, Data, Disclosure of Information, Damage[3]. We might want to add:

(1) to date, attackers have only been observed to conduct their attacks during times that match their historical profile <insert details about historical profile>.

(2) the toolkits seem primitive, perhaps a brute force method.

(3) damage to this point seems largely reputational.

The aforementioned examples is of highly speculative nature. The annotations do not provide evidence for their contextual data, is subjective and is personal (i.e. it is a personal reflection of the situation). An argument against such annotations would be that, even here, highly speculative information may be wrong, and, moreover, the subjective information may be subject to misinterpretation. The purpose of these annotations is to describe

thoughts-in-progress about nuances, and, as long as the reader is aware of this contextual information, that should be enough for them to not take this information at face value: the annotations are intended as mental notes to the person writing them not necessarily be notes to be shared across analysts. For more evidence when describing nuances in in annotations, any metric the analyst or academic wishes to use could be selected. As a best-practice, a justification of the metric should be in place. For instance, if precision, accuracy (statistical bias or observational error) or sensitivity are concepts that matter from a measurability perspective, then those metrics can be used (although these are simple examples – any metric can be used). Precision here, for instance, referring to a *Positive Predictive Value*.

We argue that the use of annotations may amplify cognition of analyst analysis of attacks and can help in asking better questions during identification of unknown aspects of attacks faster, especially in the case of zero-day attacks, when information is scarce by getting many different perspectives about an attack in a single table – leaving the analyst to identify the patterns of nuances, as opposed to viewing nuances as comments only.

Annotations tables

For a more structured, evidence-based approach, a simple n × m matrix with basic colour coding can be applied.

Annotation tables are simply intended as a mechanism for which people who use attack models can append mental note information to express known issues more straightforwardly in a structured manner. They can be used as a means to write down mental notes in the form as annotation.

We describe an annotation table as a simple n × m matrix with each row listing nuances of interest in an attack, whereas the columns show aspects of an attack (e.g. vulnerability, attacker, impact – any component of an attack that exist in a model) of which we are interested in exploring in-depth (i.e. w.r.t. nuances) (Figure 2). The purpose of annotation tables is to facilitate the discussion about aspects and nuances.

We envisage this idea can be extended in four key ways:

(1) *Considering factors of nuances*: A means to describe quality of nuance (see colours in Figures 3 and 4).

(2) *Cells can be colour coded to signify scale of a factor*: Figure 3 uses low, medium and high.

(3) *Text to describe the nuance either qualitatively or quantitatively.* (see "text#" as a placeholder).

(4) *Each colour scale can represent a separate factor* (see Figure 4).

	Aspect 1	Aspect 2	Aspect 3	Aspect m
Nuance1				
Nuance2				
Nuance3				
Nuance n				

Figure 2.
An n × m matrix showing how aspects relate to nuances. An aspect here is a component of any attack model of interest to better understand its nuances

Figure 3 shows not only how aspects are listed against nuances but also how factors of nuances play a role. In the example, the severity factor represents the how much value the nuance should carry to the analyst reviewing the matrix. We assume that the intrinsic importance of nuance, aspects and factors are the same; the severity in this example refers to how much the nuance is deemed to matter for that aspect.

Note the use of a sequential factor colouring scheme as opposed to a divergent or categorical one. This not only allows the nuances to be listed in order but could also be computed straightforwardly (see future work in the conclusion) but also allows us to perform basic arithmetic operations on the annotation table (should that be desirable. A use case may be to conduct a difference operation between two tables generated by two separate analysts to identify the differences of opinion between two separate analysts who are investigating the same attack).

	Aspect 1	Aspect 2	Aspect 3	Aspect m
Nuance1	Text1	Text5	Text9	text13
Nuance2	Text2	Text6	Text10	text14
Nuance3	Text3	Text7	text11	text15
Nuance n	Text4	Text8	Text12	Text16

Legend:
Severity
High
Medium
Low

Figure 3.
An n × m matrix showing – the text in each cell describe the nuance at each aspect (either qualitatively or quantitatively), whereas the colour represents a factor of the nuance

	Impact			Vector		
Measurability	text1	text6	text11	text31	text36	text41
Precision	text2	text7	text12	text32	text37	text42
Accuracy	text3	text8	text13	text33	text38	text43
Transparency	text4	text9	text14	text34	text39	text44
Nuance n	text5	text10	text15	text35	text40	text45

	Motives			Attribution		
Measurability	text46	text51	text56	text61	text66	text71
Precision	text47	text52	text57	text62	text67	text72
Accuracy	text48	text53	text58	text63	text68	text73
Transparency	text49	text54	text59	text64	text69	text74
Nuance n	text50	text55	text60	text65	text70	text75

Legend:

Severity	Confidence	Variance
High	Observed (High)	High
Medium	Inferred (Medium)	Medium
Low	Speculated (Low)	Low

Note: This table also shows a more complete example of the annotation table appearance

Figure 4.
Each column can be split into sub-columns to highlight aspect factors, if analysts want to express factors about nuances, such as confidence in observations and variance in the observations made

Finally, if more factors are desirable (particularly to show patterns in observations of data), it would include them. It is important to note that what factors are to be used should be determined by the analyst, and clear definitions of what each factor, property, aspect and nuance should be included, either in a legend or other documentation. Examples of possible factors may include:

- *Confidence* (in our information) – i.e. "how much evidence is there to be able to support the nuance?"
- (*Informal) Variance* – i.e. "how much inconsistency in our evidence have we seen?"
- *Severity* – i.e. "how important do we consider the nuance to be in the context of our attack model?"

Conclusion

A number of models exist that attempt to describe cyberattacks with varying degree of granularity. In this paper, we have proposed an approach that aims to help academics and analysts append detailed information about attacks on an ad hoc basis. Annotations allow mental notes that may not fit a model otherwise be included. These mental notes can still be considered: either in an analyst's decision-making or while when limited resources are available before making any decision (from using attack models).

The paper outlined the basic uses of nuances as a means to describe subtle differences between attacks. We outlined commonalities across a broad range of existing attack models in the effort to help us better understand how subtle differences in attacks can matter and express them as annotations in any attack model. This is done in the effort to help analysts be able to express subtle details in attacks that may not fit a model that might be important to make a note of without compromising or replacing an existing model or framework they are already using. It was argued that the use of such annotations can help analysts ask more well-informed questions during identification of unknown aspects of attacks faster, especially when facing zero-day attacks when information is otherwise incomplete.

We have considered conceptual use cases to demonstrate how the annotation method can be used in real-world examples. In the future, we intend to investigate how this annotation approach can be extended further. Specifically, we intend to investigate:

- *how annotation tables can be created computationally* – it may be possible to produce annotation tables from empirical evidence in an automated manner.
- *how researchers and analysts may respond to using this approach* – it will be necessary to assess how useful annotations and nuances are in the real world.
- *how complementary this approach actually is* – We currently assume our approach is complementary and showcase some conceptual examples of why this may be the case. However, it will be necessary to present analysts with real-world attacks.
- *whether biases emerge from using such methods*.
- *whether the approach can be used in cause and effect models, risk models and threat models.* In this paper, we have almost exclusively focused on attack models. It would be useful to expand the uses of our approach to other related models.
- *how our approach can help evaluate the effect of attack actions on the system performances* – for instance, by conducting sensitivity analysis (in attacker resource restrictions), we may be able to infer how much net effect nuances may have on the system being attacked.

Another issue that needs to be addressed (more generally) in the future relates to incompatibility of definitions/terminology or classifications. These emerges once an analyst begins to place one attack model in the context of other attack models. In Howard and Longstaff's model pre-dates Mitre's CVE and CWE standards, and their vulnerability definition appears to contradict Mitre's definition. This monograph does not set out to identify all contradictions across publications, but the point made here is that ambiguity is bound to happen – and an analyst stance on the issue should be addressed. From the paper, it appears the term vulnerability is used generically stating: "*vulnerability - a weakness in a system allowing unauthorized action*" Howard & Longstaff's (1998), whereas Mitre's CVE and CWE efforts make an explicit distinction between vulnerabilities and weaknesses, stating on the CWE website[1]:

> Software weaknesses are errors that can lead to software vulnerabilities. A software vulnerability, [. . .], is a mistake in software that can be directly used by a hacker to gain access to a system or network.

Note

1. https://cwe.mitre.org/about/faq.html#A.2

References

Abbott, R.P., Chin, J.S., Donnelley, J.E., Konigsford, W.L., Tokubo, S., Webb, D.A. and Linden, T.A. (1976), "Security analysis and enhancements of computer operating systems: the RISOS project", *Technical Report NBSIR 76-1041*, Institute for Computer Sciences and Technology, National Bureau of Standards.

Agrafiotis, I., Nurse, J.R., Buckley, O., Legg, P., Creese, S. and Goldsmith, M. (2015), "Identifying attack patterns for insider threat detection", *Computer Fraud & Security*, Elsevier, Vol. 2015 No. 7, pp. 9-17.

Almasizadeh, J. and Azgomi, M.A. (2009), "Intrusion process modeling for security quantification", International Conference Availability, Reliability and Security, *IEEE*, *Fukuoka*, pp. 114-121.

Barnum, S. (2008), *Common Attack Pattern Enumeration and Classification (CAPEC) Schema Description*, Mitre Coorperation.

Bisbey, R. and Hollingsworth, D. (1978), *Protection Analysis Project Final Report*, Information Sciences Institute, University of Southern California, Marina Del Rey, CA.

Bishop, M. (1995), "A taxonomy of Unix system and network vulnerabilities", *Technical Report, Technical Report CSE-95-10*, Department of Computer Science, University of California, Davis.

Bishop, M. and Bailey, D. (1996), *A Critical Analysis of Vulnerability Taxonomies*, Department of Computer Science, University of California, Davis.

Cohen, F. (1997), "Information system attacks: a preliminary classification scheme", *Computers & Security*, Elsevier, Vol. 16 No. 1, pp. 29-46.

Deng, R., Xiao, G. and Lu, R. (2017), "Defending against false data injection attacks on power system state estimation", *IEEE Transactions on Industrial Informatics*.

Grance, T.K. (2006), *Common Vulnerability Scoring System*, IEEE Security & Privacy.

Happa, J. and Fairclough, G. (2017), "A model to facilitate discussions about cyber attacks", *Ethics and Policies for Cyber Operations*, Springer International Publishing, pp. 169-185.

Happa, J., Fairclough, G., Nurse, J.R., Agrafiotis, I., Goldsmith, M. and Creese, S. (2016), "A pragmatic system-failure assessment and response model", *International Conference on Information Systems Security and Privacy*, *SCITEPRESS Digital Library*.

Homer, J., Ou, X. and Schmidt, D. (2009), "A sound and practical approach to quantifying security risk in enterprise network", *Kansas State University Technical Report*.

Howard, J.D. and Longstaff, T.A. (1998), *A Common Language for Computer Security Incidents*, Sandia National Laboratories.

Hutchins, E.M., Cloppert, M.J. and Amin, R.M. (2011), "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains", Leading Issues in Information Warfare & Security Research.

Janse van Rensburg, A., Nurse, J.R. and Goldsmith, M. (2016), "Attacker-parametrised attack graphs", *SECURWARE International Conference on Emerging Security Information, Systems and Technologies*, *Nice*.

Kandias, M., Mylonas, A., Virvilis, N., Theoharidou, M. and Gritzalis, D. (2010), "An insider threat prediction model", Conference on Trust, Privacy and Security in Digital Business, *Springer Berlin Heidelberg*, pp. 26-37.

Landwehr, C.E., Bull, A.R., McDermott, J.P. and Choi, W.S. (1994), "A taxonomy of computer program security flaws", *ACM Computing Surveys (CSUR)*, pp. 211-254.

Legg, P.A., Moffat, N., Nurse, J.R., Happa, J., Agrafiotis, I., Goldsmith, M. and Creese, S. (2013), "Towards a conceptual model and reasoning structure for insider threat detection", Managing Insider Security Threats (MIST).

Madan, B.B., Goševa-Popstojanova, K., Vaidyanathan, K. and Trivedi, K.S. (2004), "A method for modeling and quantifying the security attributes of intrusion tolerant systems", *Performance Evaluation*, Vol. 56 Nos 1/4, pp. 167-186.

Magklaras, G.B. and Furnell, S.M. (2005), "A preliminary model of end user sophistication for insider threat prediction in IT systems", *Computers & Security*, Vol. 24 No. 5, pp. 371-380.

Mann, D.E. (1999), *Towards A Common Enumeration of Vulnerabilities*, Workshop on Research with Security Vulnerability Databases, Purdue University, West Lafayette, IN.

Martin, R.A. (2007), *Common Weakness Enumeration*, Mitre Corporation, McLean, VA.

NIST (2004), *Standards for Security Categorization of Federal Information and Information Systems*, Computer Security Division Information Technology Laboratory, National Institute of Standards and Technology (NIST).

Obrst, L., Chase, P. and Markeloff, R. (2012), *Developing an Ontology of the Cyber Security Domain*, STIDS, pp. 49-56.

Roscoe, B. and Goldsmith, M. (1997), *The Perfect spy for Model – Checking Crypto – Protocols*, Rutgers University, Piscataway, NJ.

Ryan, P., Schneider, S.A., Goldsmith, M., Lowe, G. and Roscoe, A. (2000), *The Modelling and Analysis of Security Protocols: The CSP Approach*, Addison-Wesley Professional, Boston, MA.

Simmons, C., Shiva, S., Dasgupta, D. and and Wu, Q. (2009), "AVOIDIT: a cyber attack taxonomy", *Technical Report CS-09-003*, University of Memphis.

VERIS (2015), "Vocabulary for event recording and incident sharing", available at: http://veriscommunity.net/

Wang, L., Liu, A. and Jajodia, S. (2006), "Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts", *Computer Communications*, Vol. 29 No. 15.

Zhang, H., Cheng, P., Shi, L. and Chen, J. (2015), "Optimal denial-of-service attack scheduling with energy constraint", *IEEE Transactions on Automatic Control*, Vol. 60 No. 11.