An Empirical Evaluation of the Effectiveness of Attack Graphs and Fault Trees in Cyber-Attack Perception

Harjinder Singh Lallie*, Kurt Debattista†, Jay Bal†
*Cyber Security Centre, WMG, University of Warwick, UK
†WMG, University of Warwick, UK
{HL, K.Debattista, Jay.Bal}@warwick.ac.uk

***Abstract*—Perceiving and understanding cyber-attacks can be a difficult task. This problem is widely recognised and well documented, and more effective techniques are needed to aid cyber-attack perception. Attack modelling techniques (AMTs) - such as attack graphs and fault trees, are useful visual aids that can aid cyber-attack perception; however, there is little empirical or comparative research which evaluates the effectiveness of these methods. This paper reports the results of an empirical evaluation between an adapted attack graph method and the fault tree standard to determine which of the two methods is more effective in aiding cyber-attack perception. An empirical evaluation (*n=63*) was conducted through a $3 \times 2 \times 2$ factorial design. Participants from computer-science and non computer-science backgrounds were divided into an adapted attack graph and fault tree group and then asked to complete three tests which tested the ability to recall, comprehend and apply the attack modelling technique. A mean assessment score (*mas*) was calculated for each test.**

The results show that the adapted attack graph method is more effective at aiding cyber-attack perception when compared with the fault tree method ($p<0.01$). Participants that have a computer science background outperformed other participants when using both methods ($p<0.05$). These results indicate that the adapted attack graph method can be an effective tool for aiding cyber-attack perception amongst experts. The study underlines the need for further comparisons in a broader range of settings involving additional techniques, and suggests several suggestions for further work.

***Index Terms*—Cyber-attack, attack modelling, cyber-visualisation, attack graph, fault tree, attack tree**

I. Introduction

Recent well known attacks such as the *WannaCry* ransomware attack [1] have highlighted problems relating to the ability of decision makers to perceive cyber-security - in particular cyber-attacks [2]. The problem of perception in the present study relates to the ability to assess and understand the sequence of events that led to a cyber-attack. This problem applies to expert computing/IT practitioners - who have to make real time incident response decisions as well as non-experts, such as CEOs - who are part of the decision-making process and may not fully understand the technical implications. CEOs have low cyber-literacy levels - with 91% reporting problems with interpreting cyber-security reports [3]. Decision makers can be poorly advised and under-prepared to tackle cyber-security challenges [4] and don't necessarily possess the key knowledge and understanding to drive action [5, 2]. Consequently, there is a view that decision makers consider the cyber-security domain to be perceptually inaccessible [6].

Better techniques are required to aid the understanding of cyber-security among such audiences so as to enable a more effective understanding of risk [7] and better decision making. Attack modelling techniques (AMT) are a method of modelling and visualising the sequence of events that enable a successful cyber-attack on a host or network. AMTs allow analysts to understand the underlying susceptibility of a host or network to a cyber-attack - and in doing so, to identify weaknesses and vulnerabilities in a host or network. Furthermore, these techniques permit decision-makers and other non-experts to form an understanding of potential underlying vulnerabilities and means of preventing them.

Although AMTs serve a useful academic and practical purpose, there has been little or no research on the measurable cognitive impact of AMTs - on both experts and non-experts. This situation exists for many other visual information flow models as well. For example, there exist few if any controlled evaluations comparing data flow charts or status diagrams with other methods of expressing information flow. Quite often there is a presumption that a particular visual method is the 'best' mechanism for expressing information flow. This is an important research area, and this paper contributes towards addressing this research gap by presenting the results of an empirical evaluation of attack graphs and fault trees as methods of aiding cyber-attack perception.

Although numerous research papers have deployed attack graphs to represent cyber-attacks, there is no standardised attack graph method. More than fifty self-nominated methods have been adopted in the academic literature to display an attack graph - each of which represents the key aspects of a cyber attack in a subtly different way. The domain space demands the proposal of a validly formulated attack graph method which fully represents the fundamental cyber-attack constructs. Fault trees on the other hand are defined by an international standard [8] and have been used to describe cyber-attacks.

This study compares an adapted attack graph (*aag*) with the fault tree method to determine which of the two methods is more effective at aiding cyber-attack perception. While the two methods bear some conceptual

similarities, they differ in terms of the symbol construction and data flow, and the study attempts to outline which of the two sets of visual structures is more effective at aiding cyber-attack perception. The term 'effectiveness' in the present context refers to the ability of a participant to respond correctly to a question requiring the interpretation of the visual syntax of a given AMT. The study finds that the *aag* method is more effective than the fault tree method ($p < 0.01$) at aiding cyber-attack perception. Furthermore, participants with some computer science knowledge demonstrated a higher ranking when using *aag* in comparison with those using the *fault tree* method ($p < 0.05$) signifying that the *aag* method can be effective in increasing cyber-attack perception amongst experts.

The novelty and contributions of the work presented herein are as follows, the research:

1) Demonstrates that the *aag* can be used as a more effective method of aiding cyber-attack perception amongst expert audiences when compared with fault trees
2) Proposes a methodology that enables researchers to measure the effectiveness of visual information flow methods
3) Outlines initial efforts towards defining a standardised attack graph method.

The rest of this paper is organised as follows. Section II outlines related work. Section III introduces the *aag* method and fault tree methods. Section IV outlines the experimental framework and outlines the methods used, the demographics of the participants and the procedure followed by the participants. Section V provides an analytical overview of the results - particularly highlighting the tests applied to the results to highlight the statistical significance. Section VI explains the results and outlines areas for further research.

II. Background

The ability to present security problems in a manner that enables decision makers to effectively perceive the problem is a key security challenge [9]. The visualisation of complex knowledge and information structures helps learners to better perceive complex concepts. Visual methods such as AMTs have considerable value in aiding cyber-attack perception [10, 11]. Fithen et al., [12] outline the benefits of attack visualisation by arguing that AMTs enable non-experts to better understand and interpret attack models with little reference to logical models. Roschke et al., [13] propose that AMTs remove the intellectual burden from security experts who have to understand and evaluate numerous potential options. Effective visual representations enable security personnel to achieve a quick understanding of the problem domain.

A lot of the research into cyber-attack perception appears to have focussed on aiding the cyber situational awareness of experts [14, 15, 16], However, there is a recognition - expressed largely in the business press, that there is a wide-spread problem of cyber-attack perception amongst non-experts [5, 6] and a recognition that AMTs can aid the assessment and understanding of cyber-attacks. However, there is a dearth of academic research into the benefits of AMTs on cyber-attack perception.

Opdahl and Sindre [17] performed a qualitative evaluation which compared the attack tree method with *misuse cases* in aiding practitioner perception in threat identification. Opdahl and Sindre set out to evaluate the effectiveness of the two techniques - measured as the number of threats found by participants in two sample scenarios, the number/types of threats found by participants; and participant perceptions of the two techniques. The research presented three key findings. The attack tree method was more effective in aiding threat identification when compared with the misuse case method, participant perceptions of two techniques were similar, however, participant perception did not correlate with the performance of the participant in using the selected technique.

The contribution by Flaten and Lund [18] attempted to understand whether attack trees could improve an expert's understanding of a cyber-security threat. The research harnessed the views of two cyber-security experts who answered a number of questions regarding two attack scenarios - both presented as attack trees. The study found that attack trees are not suited for aiding cyber-attack perception in cyber-security experts. Other than these two studies there appears to have been no other research focussing on the cognitive benefits of AMTs.

The key differences between the approaches of Opdahl and Sindre, and Flaten and Lund and the approach presented herein are as follows. The contribution by Flaten and Lund involved a small number of participants (*n=2*) and considered one AMT. The present study adopts a quantitative approach with 63 participants. The two methods compared by Opdahl and Sindre (attack trees and misuse cases) are considerably different in terms of their visual syntax. Such a comparison requires a careful analysis and consideration of the impact that differences in the two techniques could have on participants.

The present method compares two conceptually similar techniques. Such an approach allows for a focussed analysis of the two techniques and enables a refined understanding of the differences between the two techniques.

III. Attack Modelling Techniques

AMTs represent cyber-attacks by using semantic methods (formal languages) and/or visual syntax [72] in the form of a tree/graph/net. The visual representation of an attack - referred to herein as 'visual syntax', utilises symbolic modes of expression to visualise one or more of the three *fundamental cyber-attack constructs* which are: the preconditions/postconditions of a cyber-attack (also referred to as a 'status'); exploits (also referred to as an 'event'); and precondition logic. Examples of these are given in Figure 1 where a precondition is represented as a box, an exploit as an ellipse and precondition logic by the presence or absence of an arc connecting two edges.

An exploit can be represented as a tuple of the form: (h_s, h_d, v) wherein source host h_s can exploit a vulnerability v which exists on a destination host h_d [73].

A precondition is one or more host/system statuses that must exist for an exploit to be successful. The postcondition is the state of the host/system once the exploit has

Category	Sub Category	References
Attack Tree	Attack trees	[18, 19, 20, 21, 22, 23, 24, 25]
	Defense trees	[26, 27, 28, 29]
	Threat trees	[30, 31, 32]
	Fault trees	[33]
	Attack nets	[20, 34, 33, 35]
	Threat nets	[36]
	Protection Trees	[37, 38]
	Vulnerability Trees	[39, 40, 41, 42, 43]
Attack Graph	'General' Attack graphs	[44, 45, 46] and [47]
	Alert Correlation Graphs	*Alert Correlation Graphs*: [48, 49, 50, 24, 51, 13, 52]; *Hyper-Alert Correlation Graphs*: [53]
	Privilege graphs	[54], [55], [56]
	Vulnerability graph	*Exploitation graphs*: [57, 58, 59]; *Exploit dependency graphs*: [44, 60]; *Exploit oriented graphs*: [61]; *State enumeration attack graphs*: [61]; *Dependency attack graph*: [61]
Others	Petri Net Based Models	[20, 34, 33]
	Influence diagrams	[62, 63, 64, 65, 66, 67], *extended influence diagram* [68, 69]
	Kill chains	[70]
	Diamond model	[71]

Table I: Attack Modelling Techniques

been applied. Quite often, the postcondition is in itself the precondition to another exploit.

Precondition logic is the conjunctive/disjunctive (AND/OR) relationship between preconditions. Quite often, one or more conditions must exist for an exploit to be successful.

A *conjunctive* precondition relationship requires all the connected preconditions to be fulfilled for the exploit to be successful. An example of this is provided in Figure 1 wherein two preconditions must exist for the *sshd_bof(3,1)* exploit to be applied: *sshd(3,1)* **and** *user(3)*. A *disjunctive* precondition relationship requires for any one or more of the connected preconditions to be fulfilled for the exploit to be successful. An example of this is provided in Figure 1. In this example, any one of two exploits (*sshd_bof(1,2)* **or** *sshd_bof(3,2)*) must be applied to gain *user(2)* status on the target machine. It is not common for precondition logic to be represented in a given AMT, however, it is important for an analyst to know the relationship between the conditions. Addressing just one of multiple conjunctive preconditions - which an exploit relies upon, can act as a suitable mitigation strategy.

Numerous AMTs have been proposed in the academic literature. A number of the more popular AMTs are highlighted in Table I and include: attack trees [19, 20], defence trees [26, 27, 28], privilege graphs [56], exploitation graphs or e-graphs, [57, 59], fault trees [33], Petri Net Models [20, 34, 33], alert correlation graphs [48, 49, 51, 13, 52] and influence diagrams [62, 63, 64]. Of these methods, fault trees and Petri nets are the only attack modelling techniques to be defined in an international standard [8, 74]. Some of these methods - for example fault trees and Petri nets, were not designed to represent cyber-attacks, but have been used in the literature to describe cyber-attacks.

A. Selection of AMT

The present study compares an adapted attack graph method with the fault tree method. Attack graphs and fault trees were chosen for the empirical comparison for reasons of *functionality* and because they are widely *academically accepted*.

Functionality: One of the key factors in selecting the two methods to be compared was whether the method is able to represent the fundamental cyber-attack constructs without requiring significant modification. Of all the methods outlined in Table I, attack graphs (particularly those described by Barik and Mazumdar [47] and Ghosh and Ghosh [75]) and fault trees are able to represent these constructs with very little or no modification. Methods such as kill chains [70] and the diamond model [71] are not designed to represent the fundamental cyber-attack constructs. Petri nets were designed to enable experts to understand information flow and control in systems and are particularly useful in describing systems that exhibit concurrency and asynchronous behaviour [76]. They are useful for experts but do not lend themselves to easy perception by non experts. It would be unfair to compare such a system with a method that is much more visually expressive.

Academic acceptance: Attack graphs and fault trees are an accepted and popular form of attack representation amongst the academic community and have been applied in multiple wide ranging scenarios. Methods such as kill chains and the diamond model are popular amongst the business community but not necessarily amongst the academic community.

The fault tree design is governed by an international standard [8] which defines numerous symbols. A small subset of these symbols are ideal for use in describing a cyber-attack scenario - making the fault tree method ideal for comparison.

Both methods are described more fully in section III-C1 and III-C2.

B. Visual Structures

The adapted attack graph *(aag)* method is based on the attack graph representations by Noel et al., [44], Foo et al., [45], Wang et al., [46], and Barik and Mazumdar [47]. The adapted attack graph method is demonstrated in Figures 1 (right), 2 (right) and 3 (right) and the following key explains the icons used therein.

- Preconditions/postconditions: rectangle
- Exploits: ellipse
- Conjunction: arc connecting preconditions
- Disjunction: Absence of an arc connecting preconditions
- Grey rectangle: overall attack goal

The two models are conceptually similar. However, there are some fundamental differences in their visual

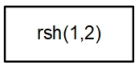
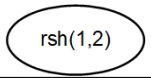
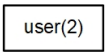
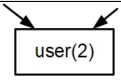
Construct	Fault tree	Adapted Attack Graph
Exploit		
Precondition		
And		
Or		
Event Flow	Bottom to top	Top to Bottom

Table II: Visual Syntactic Differences between the Fault Tree and Adapted Attack Graph Methods

syntax structures which are likely to render differences in cognitive perception. These differences are presented in Table II and can be summarised as follows.

- **Symbol usage** Both methods utilise two symbols to represent exploits and preconditions. *aag* utilises an ellipse for an exploit and a rectangle for a precondition. For the fault tree method it is the other way around.
- **Representation of precondition logic.** The fault tree method utilises two symbols to represent precondition logic. The *aag* method utilises the presence or absence of an arc to represent conjunction and disjunction respectively
- **Symbol count.** Correspondingly, the total number of symbols used in the fault tree method is four, in the *aag* method it is two
- **Event flow.** Events flow upwards in a fault tree (or rather conceptual reasoning starts at the top) whereas in the *aag* method (and in attack graphs in general) they flow downwards

Some of these simple differences such as the inclusion of a specific symbol to represent an *OR* condition - thereby increasing the total symbol count, and the direction of information flow are likely to impact the results.

C. Attack Graphs and Fault Trees

1) Attack Graphs: Attack graphs are possibly the most popular AMT - particularly in the academic literature. An attack graph is a mathematical abstraction of attack paths that might be perpetrated against a given system [77, 78]. The graph comprises of nodes which represent exploits/attacks/events and edges which represent a change of status.

The nodes in the graph can represent a range of elements such as:

- An **exploit** that has been or could be applied to the given node [61, 79, 80]
- An **event** such as an *access violation* [81] or a remote server exploit [82, 51] which forms the necessary stages in an attack
- A **status** or condition attained by an exploit [83]. Examples of this include preconditions/postconditions. Quite often, preconditions/postconditions are defined as *privileges* [55, 56]

Edges in an attack graph can be directed - to represent specific transitions, or undirected - to represent a general connection between two nodes and generally represent the perpetration of an exploit. However, edges can also represent: *actions* [84], *preconditions* - where a precondition edge $e = (a, s)$ exists if a is a precondition of s [57], or *vulnerabilities* - where a vulnerability edge represents a vulnerability that a perpetrator could exploit [85, 78].

2) Fault Trees: Both fault trees and attack trees find their origins in decision trees which aid decision making through the representation of three elements: a *decision node, edge/branch* and *leaf*. Although decision trees have been applied to a computer/cyber security context [86, 87, 88, 89, 90, 40], decision trees were not designed to aid visual consumption, and it is probably for these reasons that methods such as attack trees and fault trees were developed. Schneier recast decision trees in the form of attack trees [19] and may have been influenced by the fault tree method [40].

Attack trees and fault trees are acyclic directed graphs which outline important events and conditions. Events lead to a goal condition which is referred to as an *attack goal* in an attack tree [19], and an *undesirable condition* in the case of a fault tree [91].

The symbolic representation of fault trees was standardised by the IEC in 1990, [8], the European Cooperation for Space Standardization [92] and then by the British Standards Institute [93]. However, although the fault tree structure is standardised, there is no agreed method of representing attack trees, and - like attack graphs, there exist numerous - subtly different versions of attack trees.

Fault trees are the most visually expressive AMT because they utilise a wide range of standard symbols to express elements of an attack. Fault trees are used in a number of industries such as in the aerospace industry [94, 95, 96, 97], radioactive waste disposal [98], the automotive industry [99, 100] and in the analysis of failure in computer systems [101, 102, 103]. Although the fault tree standard is a generic standard (not particularly focussing on cyber-security as a target domain), more recently, fault trees have become a popular means of representing cyber-attack [40, 104, 105].

Example fault trees are given in Figures 1, 2 and 3. These examples demonstrate the use of conjunction and disjunction with specific symbols, preconditions as ovals and exploits as rectangles.

3) Syntactic Structure: Attack graphs and fault trees comprise of two fundamental elements represented as graph data structures of the form: $G(V; E)$ [108] which comprises of nodes and edges: $e \in E$ which represent relationships between the nodes. An attack graph/fault tree is a tuple $G = (S, \tau, S_0, S_s, L, E)$ where:

- S is a finite set of states,
- $\tau \subseteq S \times S$ is a transition relation
- $S_0 \subseteq S$ is a set of initial states
- $S_s \subseteq S$ is a set of success states – for example obtaining root or user privileges on a particular host

Test	Lower order cognitive skill	Test Description	Scenario reference	Sample question
1	Knowledge Recall	Multiple Choice select one answer	scenario 1 [47]. 4 questions	"What are the necessary exploits for an attacker to be able to achieve *user* access on *host* 2", also see Figure 4
2	Comprehension	Select correct scenario from a heatmap	scenario 2 [75] 4 questions	"Study the image below and select the **exploit(s)** which result in the attacker gaining **user access** status on host 2."
3	Application	Multiple Choice, read scenario and select one from three heat maps	scenario 3 [106, 107] 4 questions	Study the figure below and select the figure that most accurately describes the following scenario: "The *stuxnet* virus is installed when a new *services.exe* file and a new *s7otdbxdx.dll* file are installed. Before these can be installed, the following preconditions must be met. The target has to have the *RPC vulnerability*, the target has to be running the *Step7* application, and the target has to be a *Stimatic PLC*"

Table III: Description of Study Scenarios and Tests

- $L : S \rightarrow 2^{AP}$ is a labelling of states with a set of atomic propositions (*AP*)
- E is a finite set of exploits which connect the transition between two states

IV. Empirical Evaluation

A. Design

The empirical evaluation uses three independent variables (*test, AMT, background*) and one dependent variable (*mean assessment score - mas*).

The study aims to evaluate whether any of the three independent variables (*test, AMT* and *background*) influence cyber-attack perception.

The *test* is a within participant independent variable which represents the three questions asked to participants. Cyber-attack assessment and understanding was measured using a three phase test which demands the demonstration of increasingly complex cognitive skills. Table III provides an overview of its characteristics. Each test corresponded to one of the three lower levels (knowledge, comprehension and application) of Bloom's Taxonomy of educational objectives [109]. Bloom's taxonomy has been used in numerous academic fields to assess both lower and higher order cognitive skills. The result of the test is measured as a mean assessment score *mas* - the dependent variable.

The wording of the questions was carefully prepared to correspond with guidance provided on how to frame questions and utilise keywords so as to correspond with levels within the taxonomy (for example, see [110]). The test framework - including question samples, are provided in Table III.

The *background* independent variable is divided into two groups: *cs* and *oth*. The *cs* group have a computer science background. These participants have either studied computer science at undergraduate level, or have more than five years of work experience in the computing industry. The *other* group are all other participants.

There are two *AMT* groups: *aag* and *ft* - each presented with an adapted attack graph or fault tree scenario respectively.

The following two hypotheses were established:

$H1_1$ The selection of *AMT* influences response to *mas*

$H2_1$ The selection of *background* influences response to *mas*

The *AMT* and *background* form the two between-participant independent variables and the design can be described as: *3 (test)* $\times$ *2 (AMT)* $\times$ *2 (background)* yielding 12 different conditions. A two-way repeated-measures ANOVA test was used to determine the significance of the results.

The same scenarios were used and all participants were asked the same questions in the same sequence. Collectively, these formed the control variables.

B. Materials

Each test utilised a corresponding attack scenario which was converted into an attack graph and corresponding fault tree. Attack scenario 1 (Figure 1) and 2 (Figure 2) were based on the fictional attack graphs produced by Barik and Mazumdar [47] and Ghosh and Ghosh [75] respectively. These scenarios were selected as they are published scenarios and have small visual structures in terms of the number of cyber-attack constructs being used.

Attack scenario 3 was developed by the authors and based on the Stuxnet attack - a well-known virus attack [107, 106]. The Stuxnet attack is very complex, and the resulting attack graph/fault tree contains more than sixty cyber-attack constructs. Consequently, a small section of the Stuxnet attack - representing the exploitation of the *task scheduler* vulnerability was used for the scenario. Part of the scenario is presented in Figure 3. The comprehension context assumes that experts and non-experts are analysing and interpreting the visual syntax in particular - and not necessarily the formal syntactic definitions. In other words, the observer doesn't necessarily need to have a technical understanding of the attack.

In a professional setting, although the analysis of an attack might begin by focussing on the full attack graph, quite often, the decision maker/analyst proceeds to focus on the poignant elements of the attack. Consequently, a small section of the overall graph was considered appropriate for the present study. A further exploration of the cognitive impact and effects of studying small/large attack models by different stakeholders over longer periods of time may be useful and is considered for future work.

The third scenario is quite different in style compared with scenarios 1 and 2. Scenario 3 is presented using a simpler explanatory narrative in comparison with scenarios 1 and 2. It was considered important to be able

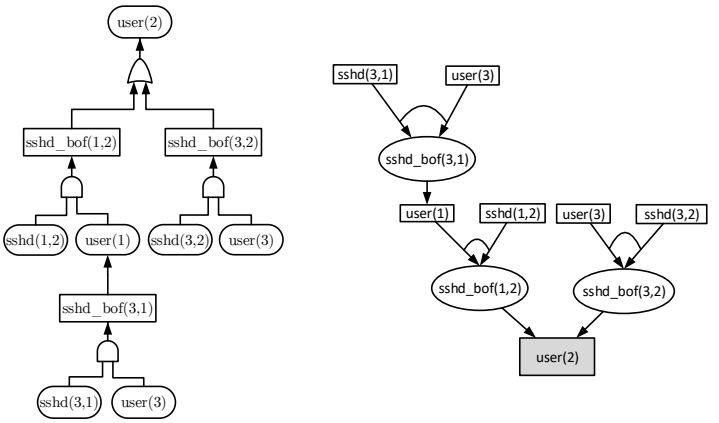


Figure 1. Cyber-attack Scenario 1 represented as a fault tree (left) and an attack graph (right) (based on Barik and Mazumdar [47]). The three fundamental cyber-attack constructs are *preconditions* - represented as boxes in the attack graph and ellipses in the fault tree, *exploits* - represented as ellipses in the attack graph and boxes in the fault tree, and *precondition logic* - *AND* represented by the presence of an arc in the attack graph and 'semi-ellipse' in the fault tree and the *OR* represented by the absence of an arc in an attack graph and an 'elliptic triangle' in a fault tree

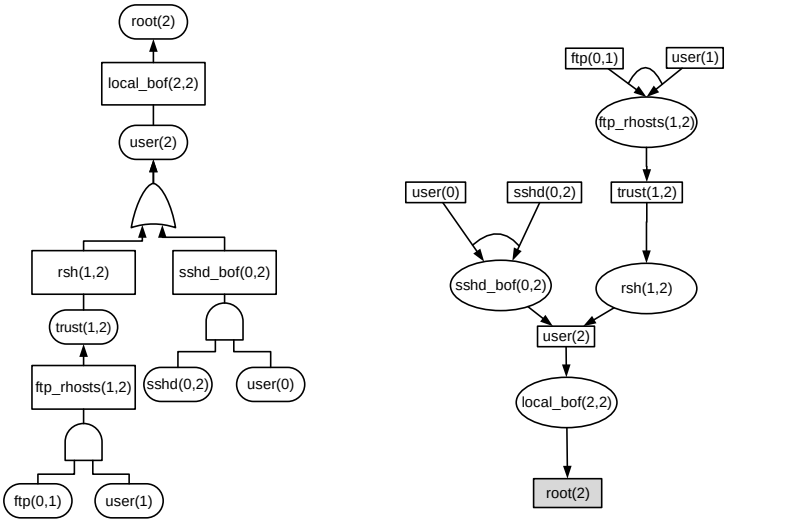


Figure 2. Cyber-attack Scenario 2, fault tree (left), attack graph (right) (Based on Ghosh and Ghosh [75])

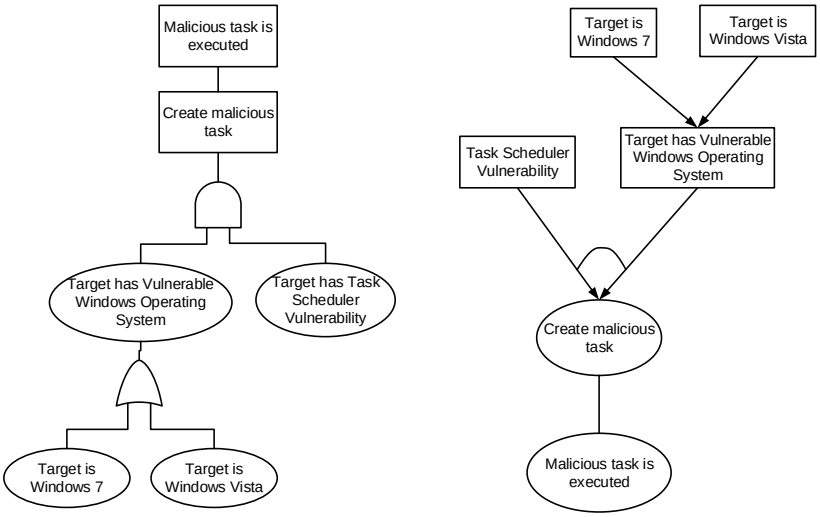


Figure 3. A Section of Cyber-attack Scenario 3, fault tree (left), attack graph (right) based on a small part of the Stuxnet attack scenario described by Falliere et al., [106]

to represent and test a portion of a 'real life' attack to understand whether this impacted participant perception.

While the data content of the three scenarios differs (formal syntax versus narrativised representation), the aim of the study is to assess the cognitive impact of the visual syntax. Consequently, one might assume that the *oth* group are more likely to score higher - in test 3 in comparison with test 1 and 2, when presented with such a representation. The results (Table V) indicate no statistical differences in the ability of non-experts to perceive the attack descriptions either when presented in formal syntactic or textually narrativised terms.

The study was configured using the Qualtrics platform [111]. The study was divided into three sections which: gathered participant consent and other data; enabled the participant to gain fundamental background information relating to the AMT being studied; and then tested participant perception through a sequence of questions (Figure 4).

C. Participants

84 participants were recruited for the study. 21 participants did not complete the study leaving 63 participants. 43 males and 20 females aged between 21 and 58 ($\overline{age} = 29$) were collected and grouped according to their *background* groups (*cs n=31*, *oth n=32*) and then further subdivided into *AMT* groups (*aag n=31*, *ft n=32*).

Participants were assigned to the *aag/ft* group randomly to avoid bias within these groups - particularly to avoid a situation where one group might understand cyber-attacks at a conceptual level better than the other.

D. Procedure

Participants were required to access the study by following a *url*. The experiment sequence was as follows:

a. The first screen provided general data regarding the study such as participant information, and required the participant to provide consent
b. The second screen gathered participant data (age, gender, experience)
c. The third screen enabled participants to study the ontology and structure of the AMT they had been assigned to.
d. Following this, participants were required to complete the test

Each question in the test required the participant to study an AMT and answer a question. Both the *aag* and *ft* group were provided with exactly the same questions. Participants were not able to revisit questions.

V. Results

Due to the large amount of data collected, the results are divided into subsections with a discussion following in section VI. The mean assessment scores for each test (*mas1, mas2, mas3*) were normalised and are outlined in Table IV. Figure 5 shows the mean assessment score (*mas*) vs *group* and *mas* vs *AMT*.

Table V highlights the mean differences between the groups. The mean difference is presented as a delta

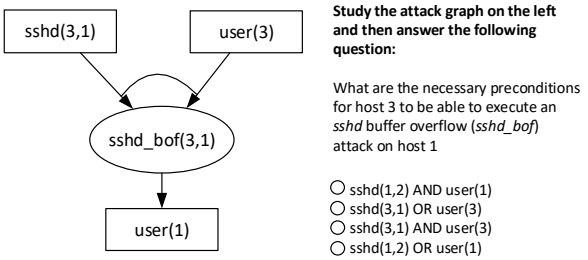


Figure 4. A Sample From the Qualtrics Based Study

(δ) value: $\delta mas(i)_{aag:ft} = (mas(i)_{aag} - mas(i)_{ft})$. The mean differences for all three tests favours the *aag* group ($\delta mas1_{aag:ft} = 0.1463, \delta mas2_{aag:ft} = 0.2475, \delta mas3_{aag:ft} = 0.0406$). Furthermore the table outlines the mean differences for the overall tests - giving an overall $\delta(mas)_{aag:ft} = 0.1448$.

These descriptive results suggest that the *aag* method is better at aiding attack perception when compared with the fault tree method. Whilst the *mas* and the mean differences appear to be significant, they require further investigation to determine statistical significance.

A. Main Effects

The main effects of *test*, *AMT* (*aag/ft*) and *background* (*cs/oth*) were analysed using a *3 (test) × 2 (AMT) × 2 (background)* mixed design factorial ANOVA for all 63 participants as outlined in section IV-A. Mauchly's Test of Sphericity indicated that the assumption of sphericity was not violated (Maulchy's test, $\chi^2(2) = 1.309, p = 0.520$).

1) Within-Participant Main Effects of test: The within-participants main effect for *test* signified that participant performance differed between tests (*F (2, 118) = 3.232, p = 0.043*). Pair-wise comparisons were performed between the three tests and statistically significant results from this analysis are presented in Table VI and discussed further for the *AMT* (section V-B1) and *background* (section V-B2) groups.

2) Between-participants main effect of AMT: Hypothesis 1 ($H1_1$) proposed that: *the selection of AMT influences response to mas*. Hypothesis 1 holds and the between-participants main effect for *AMT* revealed that the selection of *AMT* was significant, (*F(1, 59) =8.004, p=0.006*) showing that there is a distinction between the two AMTs with *aag* ranked higher than *ft*.

Table IV and Figure 5 outline differences in the performance between the *aag/ft* groups - favouring the *aag* group. ($\delta mas1_{aag:ft} = 0.1463, \delta mas2_{aag:ft} = 0.2475, \delta mas3_{aag:ft} = 0.0406$). The *aag* group demonstrated a better result overall for the test: $\delta mas_{aag:ft} = 0.1448$. This is analysed further in section V-B1 to identify differences within the *aag* and *ft* groups.

3) Between-participants main effect of background: Hypothesis 2 ($H2_1$) proposed that: *the selection of background influences response to mas*. Hypothesis 2 holds and the between-participants main effect for *background* was statistically significant (*F (1, 59) = 12.843, p = 0.001*) showing that the selection of *background* influences response to *mas* with *cs* ranked higher than *oth*.

Test		mas^1			mas^2			mas^3		
		Mean	SD	n	Mean	SD	n	Mean	SD	n
oth	*aag*	0.8353	0.2486	17	0.7582	0.2695	17	0.7059	0.3450	17
	ft	0.6267	0.3530	15	0.5020	0.3760	15	0.5667	0.4169	15
	Total	0.7375	0.3152	32	0.6381	0.3438	32	0.6406	0.3807	32
cs	*aag*	0.9357	0.1292	14	0.9386	0.1360	14	0.9107	0.1583	14
	ft	0.8294	0.2024	17	0.6718	0.2903	17	0.9265	0.1929	17
	Total	0.8774	0.1788	31	0.7923	0.2668	31	0.9194	0.1754	31
Total	*aag*	0.8806	0.2068	31	0.8397	0.2347	31	0.7984	0.2917	31
	ft	0.7344	0.2966	32	0.5922	0.3387	32	0.7578	0.3619	32
	Total	0.8063	0.2648	63	0.7140	0.3155	63	0.7778	0.3272	63
$\delta_{aag:ft}$		0.1463			0.2475			0.0406		
$\delta(cs:mas)_{aag:ft}$		0.1063			0.2668			-0.0158		
$\delta(oth:mas)_{aag:ft}$		0.2086			0.2562			0.1392		

Table IV: Mean Assessment Scores *mas*

	mas1		mas2		mas3		mas	
Delta	Mean	n	Mean	n	Mean	n	Mean	n
$\delta_{aag:ft}$	0.1463	63	0.2475	63	0.0406	63	0.1448	63
$\delta cs_{aag:ft}$	0.1063	31	0.2668	31	-0.0158	31	0.1191	31
$\delta oth_{aag:ft}$	0.2086	32	0.2562	32	0.1392	32	0.2013	32
$\delta ag_{cs:oth}$	0.1004	31	0.1804	31	0.2048	31	0.1619	31
$\delta ft_{cs:oth}$	0.2027	32	0.1698	32	0.3598	32	0.2441	32

NB: δ values are calculated as absolute differences

Table V: Mean Differences by *AMT* and *group*

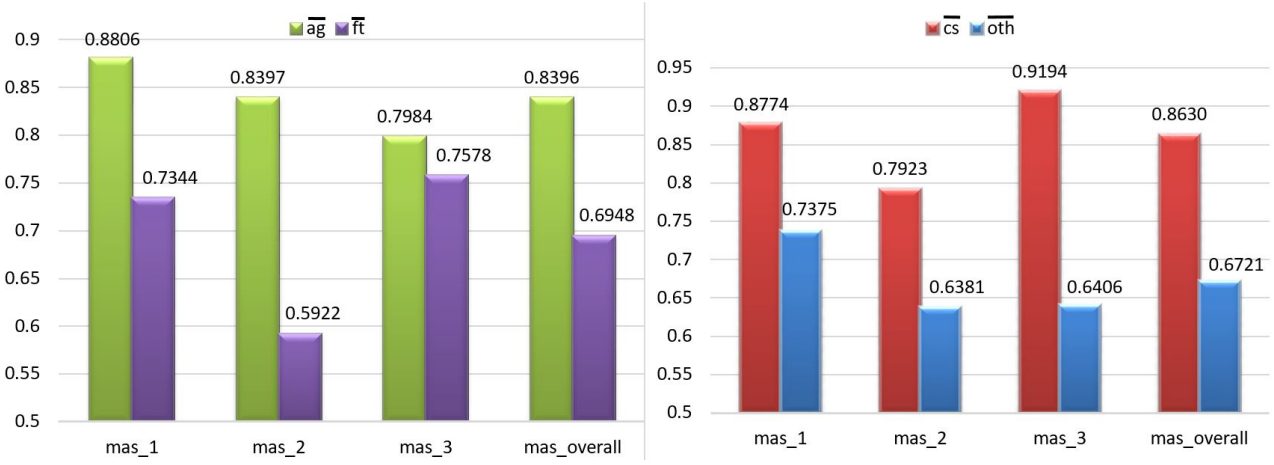


Figure 5. Overall means of *test vs AMT* (left) and *test vs group* (right)

Table V and Figure 5 highlight the mean assessment scores for the *cs/oth* groups. The Table shows that $\delta oth(mas)_{aag:ft} = 0.2013$ and $\delta cs(mas)_{aag:ft} = 0.1191$. The data outlines what appears to be a distinct difference favouring *aag* for both the *cs* and *oth* groups. This is analysed further in section V-B2 to determine if there is a statistically significant difference within the *oth* and *cs* groups.

B. Results for test, background and AMT

The main effects were analysed further to discover whether there are differences within the *background* and *AMT* groups.

1) Results for AMT: The main effect for *test* was further analysed using two: *3 (test)* $\times$ *2 (background)* ANOVAs for *aag* (n=32) and *ft* (*n=31*) to explore further effects. The between-participant effect signified that the selection of *background* influences response to *mas* (*F (1, 61)=10.321, p=0.002*).

a) Results for aag: The between-participants effect for *AMT* shows a statistically significant effect favouring *cs* ($F(1, 29) = 6.396, p = 0.017$) signifying that selection of *background* influences the *mas* score for the *aag* method. Post hoc tests using the Bonferroni correction showed that the *cs* group performed better than the *oth* group (*mean = 0.162, SD=0.064, p=0.017*). This is indicated by the overall *mas* results (Table V) which highlight

group	(i) test	(j) test	Mean difference (i-j)	std error	sig
all	1	2	0.089 (↓)	0.035	0.039
ft	1	2	0.141 (↓)	0.052	0.032
	2	3	0.160 (↑)	0.050	0.009
cs	2	3	0.113 (↑)	0.041	0.029

Table VI: Statistically Significant Within Participants Effect Results for Tests

a *mas* score favouring the *cs* group when using the attack graph method ($\delta aag_{cs:oth} = 0.1619$).

Pair-wise comparisons were performed between the three tests for the *aag* group. There were no significant differences between the tests for the *aag* group.

b) Results for ft: Similarly, the between-participants effect for *AMT* shows a statistically significant effect favouring *cs* ($F(1, 30) = 6.954, p = 0.013$) signifying that selection of background influences the *mas* score for the fault tree method. Post hoc tests using the Bonferroni correction showed that the *cs* group performed better than the *oth* group (*mean=0.244, SD=0.093, p=0.013*). This is further indicated by the overall *mas* results (Table V) which highlights a *mas* score favouring the *cs* group when using the fault tree method ($\delta ft_{cs:oth} = 0.2441$).

Pair-wise comparisons were performed between the three tests between the *ft* group. Table VI shows that the *ft* group demonstrated a reduction in *mas* of *0.141* between tests 1 and 2 (*p=0.032*) and an increase of *0.160* between 2 and 3 (*p=0.009*).

2) Results for background: The main effect for *test* was further analysed using two: *3 (test) × 2 (AMT)* ANOVAs for *cs* and *oth* to explore further effects. The between-participant effect signified that the selection of *AMT* influences the *mas* score ($F(1, 61) = 5.536, p = 0.022$).

a) Results for cs: The between-participants effect for *background* shows a statistically significant effect for *AMT* ($F(1, 29) = 5.276, p = 0.029$) signifying that the selection of *AMT* influences the *mas* score for the cs_{ft} group. Post hoc tests using the Bonferroni correction showed that the cs_{aag} group performed better than the cs_{ft} group (*mean = 0.119, SD=0.052, p=0.029*). This is further indicated by the overall *mas* results (Table V) which shows an overall *mas* favouring the cs_{aag} ($\delta cs_{aag:ft} = 0.1191$).

Pair-wise comparisons between the three tests (Table VI) show that the *cs* group demonstrated an increase in *mas* of *0.113* between 2 and 3 (*p=0.029*).

b) Results for oth: The between-participants effect for *background* did not signify a statistically significant effect for *AMT* ($F(1, 30) = 4.104, p = 0.052$) showing that the selection of *AMT* is not significant for participants in the *oth* group.

Pair-wise comparisons between the three tests for the *oth* group (Table VI) showed no significant differences between in the *mas* scores for the *oth* group.

VI. Discussion

This study investigated which of the two AMTs was more effective in aiding cyber-attack perception. The study also explored whether any demonstrable benefits of either of the two methods occurred under specific *background* grouping conditions.

Hypothesis 1 ($H1_1$) tested whether the selection of *AMT* influences response to *mas*. This hypothesis holds and the study finds that the *aag* method is better at aiding attack perception when compared with the fault tree method (*p <0.01*). A straightforward comparison of the means for both groups reveals an *mas* favouring the *aag* method for every *test*.

Hypothesis 2 ($H2_1$) tested whether the selection of *background* influences response to *mas*. This hypothesis holds and the study finds that the *aag* method can be an effective tool for aiding cyber-attack perception amongst experts. As expected, participants that have a computer science background outperformed other participants when using both methods (*p <0.05*) but demonstrated a preference for the *aag* method.

A comparison of the *mas* scores (Table IV) indicates a preference for the *aag* method compared with the fault tree method for the non-computer science group. However, this result was not statistically significant and has to be treated with caution.

Given that the three tests tested progressively advanced cognitive levels - rather than testing the same cognitive level each time, one might expect a minimum desirable result to be a stable performance across the cognitive levels. However, differences in performance (Table VI) were demonstrated between some of the tests. This was most significant for the fault tree group between tests 1 and 2 (a decrease) and test 2 and 3 (an increase). The performance increased between tests 2 and 3 for the computer science fault tree participants. There was no significant difference in the performance of the attack graph group. Notably, there was a consistent cognitive performance across the three tests for the *aag* method group with no significant decrease in performance.

Added to this, one might expect a better cognitive performance for the *oth* group for test 3 in comparison with tests 1 and 2 - given the difference in narrative style of the question. This was not the case, and as Table VI shows, there was no difference between the tests for the *oth* group.

Although the *aag* method rendered a better *mas* overall, there was a negligible difference in terms of *mas3* which tests the ability to apply the two methods. In other words, participants in both the *aag* and *ft* were able to effectively apply the methods in given scenarios. This requires further research in a more focussed study.

VII. Conclusions

The growing number of cyber-attacks and the increased need for both experts and non-experts to better understand cyber-attack leads to the requirement for better techniques and methodologies that can be used to more quickly and effectively to appraise audiences on the methods used to perpetrate cyber-attacks and the weaknesses in systems that enable such attacks to prevail.

The main contribution of this research has been to show that the *aag* method is better than the fault tree method in aiding cyber-attack perception.

A. Limitations and Future Work

The present research has revealed limitations and areas for further research that should be explored to further reason with the results presented herein.

a) Benefits on perception of non-experts: The study revealed that the *aag* method could be suitable for aiding cyber-attack perception amongst non-expert audiences - such as CEOs. Although the results in this regard were not statistically significant, this requires a further larger study to identify whether the method can indeed aid cyber-attack perception amongst non-experts.

b) Measurement of Effectiveness: The present study assessed the ability of participants to comprehended the visual syntax. The study measured effectiveness as the ability to respond correctly to a question requiring the interpretation of the visual syntax of a given AMT. This definition can be expanded further to include timeliness and attack severity.

Although the time taken to complete the study was measured, this was not analysed in the measurement of 'effectiveness' because the present study considered it more important to allow participants to apply due consideration to each question than to place the participant under time based pressure.

An alternative analysis would be to take into account the correctness of the response as a function of time. The severity of an incorrect response could also be considered important. Further research requires the development of a methodology reflecting effectiveness based on three variables: correctness, time and severity.

c) Understanding key visual syntactic factors: Further research should attempt to understand which elements of the visual structure of an AMT - described in section III-C2, are more significant in aiding cyber-attack perception. In addition, it would be beneficial to understand the effect on perception of visual structural elements such as colour, tone, line width/density/structure. In particular, further research should explore improvements to *aag* which balance the tradeoff between providing more visual information - such as the inclusion of elements such as attacker capability and/or uncertainty, whilst maintaining effective cognitive perception.

The symbol count, (including a specific symbol for OR in an FT increases the total symbol count), the direction of information flow are likely to have impacted the results. In a follow up study, we will be comparing these in a subjective evaluation.

d) Acceptability amongst practitioners: It is critical that any method of representing cyber-attacks gains acceptance amongst practitioners. Such an audience would include teacher-practitioners (lecturers who might use the method to teach cyber-attack), non-expert corporate/decision makers and cyber-security analysts. Further research should test the *aag* method with these audiences to determine the efficacy of the method in aiding cyber-attack perception in a live environment.

e) Complexity: The attack scenarios used in the study were relatively small and comprised of up to 14 symbols (Figure 2). This is not representative of complex cyber-attacks such as the Stuxnet virus [107, 106], Jeep Cherokee Hack [112] and the Sony Hack [113, 114]. Further research should examine the effectiveness of the *aag* method when presenting larger scenarios.

ACKNOWLEDGMENTS

Kurt Debattista was partially funded by a Royal Society Industrial Fellowship (IF130053)

REFERENCES

[1] BBC, "Wannacry ransomware cyber-attacks slow but fears remain," 2017. [Online]. Available: http://www.bbc.co.uk/news/technology-39920141

[2] S. Morgan, "Why CEOs Are Failing Cybersecurity, And How To Help Them Get Passing Grades," 2016. [Online]. Available: http://www.forbes.com/sites/stevemorgan/2016/05/04/why-ceos-are-failing-cybersecurity-and-how-to-help-them-get-passing-grades

[3] Tanium-NASDAQ, "The Accountability Gap: Cybersecurity & Building a Culture of Responsibility," 2015. [Online]. Available: http://q.nasdaq.com/cyber-readiness-accountability-report

[4] AT&T, "Decoding the Adversary: What every CEO needs to know about cybersecurity," 2015. [Online]. Available: https://www.business.att.com/cybersecurity/docs/decodingtheadversary.pdf

[5] KPMG, "Cyber security: a failure of imagination by CEOs," 2015. [Online]. Available: https://home.kpmg.com/xx/en/home/insights/2015/12/cyber-security-a-failure-of-imagination-by-ceos.html

[6] Odgers Berndtson, "Cyber Security - What Boards Need to Know," 2013. [Online]. Available: http://www.odgersberndtson.com/media/2253/cyber_security_-_what_boards_need_to_know_01.pdf

[7] National Cybersecurity Institute, "Cybersecurity Both Concerns and Confuses Business Exceutives," 2016. [Online]. Available: http://www.nationalcybersecurityinstitute.org/general-public-interests/cybersecurity-both-concerns-and-confuses-business-exceutives/

[8] I. IEC, "61025: Fault tree analysis," *International Electrotechnical Commission (IEC), édition), Editions Cépaduès,[12] Moraru, R., Băbuţ, G., Participatory occupational risk assessment and management: a practical guide*, 1990.

[9] J. Homer, A. Varikuti, X. Ou, and M. A. McQueen, "Improving attack graph visualization through data reduction and attack grouping," in *Visualization for Computer Security*. Springer, 2008, pp. 68–79.

[10] S. Jajodia and S. Noel, "Advanced cyber attack modeling analysis and visualization," George Mason University, Fairfax, Tech. Rep., 2010.

[11] D. Staheli, T. Yu, R. J. Crouser, S. Damodaran, K. Nam, D. O'Gwynn, S. McKenna, and L. Harrison, "Visualization evaluation for cyber security: Trends and future directions," in *Proceedings of the Eleventh Workshop on Visualization for Cyber Security*. ACM, 2014, pp. 49–56.

[12] W. L. Fithen, S. V. Hernan, P. F. O'Rourke, and D. A. Shinberg, "Formal modeling of vulnerability," *Bell Labs technical journal*, vol. 8, no. 4, pp. 173–186, 2004.

[13] S. Roschke, F. Cheng, and C. Meinel, "A new alert correlation algorithm based on attack graph," in *Computational Intelligence in Security for Information Systems*. Springer, 2011, pp. 58–67.

[14] G. P. Tadda and J. S. Salerno, "Overview of cyber situation awareness," in *Cyber situational awareness*. Springer, 2010, pp. 15–35.

[15] M. Albanese, H. Cam, and S. Jajodia, "Automated cyber situation awareness tools and models for improving analyst performance," in *Cybersecurity systems for human cognition augmentation*. Springer, 2014, pp. 47–60.

[16] J. Brynielsson, U. Franke, and S. Varga, "Cyber situational awareness testing," in *Combatting Cybercrime and Cyberterrorism*. Springer, 2016, pp. 209–233.
[17] A. L. Opdahl and G. Sindre, "Experimental comparison of attack trees and misuse cases for security threat identification," *Information and Software Technology*, vol. 51, no. 5, pp. 916–932, 2009.
[18] O. Flaten and M. S. Lund, "How good are attack trees for modelling advanced cyber threats?" *Norsk informasjonssikkerhetskonferanse (NISK)*, vol. 7, no. 1, 2014.
[19] B. Schneier, "Attack trees," *Dr. Dobbs journal*, vol. 24, no. 12, pp. 21–29, 1999.
[20] J. P. McDermott, "Attack net penetration testing," in *Proceedings of the 2000 workshop on New security paradigms*. ACM, 2000, Conference Proceedings, pp. 15–21.
[21] T. Tidwell, R. Larson, K. Fitch, and J. Hale, "Modeling internet attacks," in *Proceedings of the 2001 IEEE Workshop on Information Assurance and security*, vol. 59, 2001, Conference Proceedings.
[22] G. Helmer, J. Wong, M. Slagell, V. Honavar, L. Miller, and R. Lutz, "A software fault tree approach to requirements analysis of an intrusion detection system," *Requirements Engineering*, vol. 7, no. 4, pp. 207–220, 2002.
[23] K. Daley, R. Larson, and J. Dawkins, "A structural framework for modeling multi-stage network attacks," in *Parallel Processing Workshops, 2002. Proceedings. International Conference on*. IEEE, 2002, Conference Proceedings, pp. 5–10.
[24] X. Qin and W. Lee, "Attack plan recognition and prediction using causal networks," in *Computer Security Applications Conference, 2004. 20th Annual*. IEEE, 2004, Conference Proceedings, pp. 370–379.
[25] S. Mauw and M. Oostdijk, *Foundations of attack trees*. Springer, 2006, pp. 186–198.
[26] S. Bistarelli, F. Fioravanti, and P. Peretti, "Defense trees for economic evaluation of security investments," in *First International Conference on Availability, Reliability and Security (ARES'06)*. IEEE, 2006, pp. 8–pp.
[27] S. Bistarelli, M. Dall'Aglio, and P. Peretti, "Strategic games on defense trees," in *International Workshop on Formal Aspects in Security and Trust*. Springer, 2006, pp. 1–15.
[28] A. Bagnato, B. Kordy, P. H. Meland, and P. Schweitzer, "Attribute decoration of attack–defense trees," *International Journal of Secure Software Engineering (IJSSE)*, vol. 3, no. 2, pp. 1–35, 2012.
[29] P. Wang, W.-H. Lin, P.-T. Kuo, H.-T. Lin, and T. C. Wang, "Threat risk analysis for cloud security based on attack-defense trees," in *Computing Technology and Information Management (ICCM), 2012 8th International Conference on*, vol. 1. IEEE, 2012, pp. 106–111.
[30] A. Marback, H. Do, K. He, S. Kondamarri, and D. Xu, "Security test generation using threat trees," in *Automation of Software Test, 2009. AST'09. ICSE Workshop on*. IEEE, 2009, Conference Proceedings, pp. 62–69.
[31] I. Morikawa and Y. Yamaoka, "Threat tree templates to ease difficulties in threat modeling," in *14th International Conference on Network-Based Information Systems (NBiS)*. IEEE, 2011, Conference Proceedings, pp. 673–678.
[32] J. P. Landry, J. H. Pardue, T. Johnsten, M. Campbell, and P. Patidar, "A threat tree for health information security and privacy," in *AMCIS*, 2011, Conference Proceedings.
[33] J. Steffan and M. Schumacher, "Collaborative attack modeling," in *Proceedings of the 2002 ACM symposium on Applied computing*. ACM, 2002, Conference Proceedings, pp. 253–259.
[34] R. Lutz, "Software fault tree and colored petri net based specification, design and implementation of agent-based intrusion detection systems," 2002.
[35] I. Kotenko and M. Stepashkin, "Attack graph based evaluation of network security," in *Communications and Multimedia Security*. Springer, 2006, Conference Proceedings, pp. 216–227.
[36] D. P. Mirembe and M. Muyeba, "Threat modeling revisited: improving expressiveness of attack," in *Second UKSIM European Symposium on Computer Modeling and Simulation, EMS'08*. IEEE, 2008, Conference Proceedings, pp. 93–98.
[37] F. Tang and L. Ruan, *A protection tree scheme for first-failure protection and second-failure restoration in optical networks*. Springer, 2005, pp. 620–631.
[38] K. S. Edge, G. C. Dalton, R. A. Raines, and R. F. Mills, "Using attack and protection trees to analyze threats and defenses to homeland security," in *Military Communications Conference, 2006. MILCOM 2006. IEEE*. IEEE, 2006, Conference Proceedings, pp. 1–7.
[39] S. Vidalis, A. Jones *et al.*, "Using vulnerability trees for decision making in threat assessment," *University of Glamorgan, School of Computing, Tech. Rep. CS-03-2*, 2003.
[40] P. A. Ralston, J. H. Graham, and J. L. Hieb, "Cyber security risk assessment for scada and dcs networks," *ISA transactions*, vol. 46, no. 4, pp. 583–594, 2007.
[41] J. Holsopple, S. Yang, and B. Argauer, "Virtual terrain: a security-based representation of a computer network," in *SPIE Defense and Security Symposium*. International Society for Optics and Photonics, 2008, pp. 69 730E–69 730E.
[42] E. Tanu and J. Arreymbi, "An examination of the security implications of the supervisory control and data acquisition (scada) system in a mobile networked environment: An augmented vulnerability tree approach," 2010.
[43] D. Fall, T. Okuda, Y. Kadobayashi, and S. Yamaguchi, "Towards a vulnerability tree security evaluation of openstack's logical architecture," in *International Conference on Trust and Trustworthy Computing*. Springer, 2014, pp. 127–142.
[44] S. Noel, S. Jajodia, B. O'Berry, and M. Jacobs, "Efficient minimum-cost network hardening via exploit dependency graphs," in *Computer Security Applications Conference, 2003. Proceedings. 19th Annual*. IEEE, 2003, Conference Proceedings, pp. 86–95.
[45] B. Foo, Y.-S. Wu, Y.-C. Mao, S. Bagchi, and E. Spafford, "Adepts: adaptive intrusion response using attack graphs in an e-commerce environment," in *International Conference on Dependable Systems and Networks, 2005. DSN 2005. Proceedings.* IEEE, 2005, Conference Proceedings, pp. 508–517.
[46] L. Wang, S. Noel, and S. Jajodia, "Minimum-cost network hardening using attack graphs," *Computer Communications*, vol. 29, no. 18, pp. 3812–3824, 2006, cited by 182.
[47] M. S. Barik and C. Mazumdar, *A Graph Data Model for Attack Graph Generation and Analysis*. Springer, 2014, pp. 239–250.
[48] S. J. Templeton and K. Levitt, "A requires/provides model for computer attacks," in *Proceedings of the 2000 workshop on New security paradigms*. ACM, 2001, Conference Proceedings, pp. 31–38.
[49] F. Cuppens and A. Miege, "Alert correlation in a cooperative intrusion detection framework," in *IEEE Symposium on Security and Privacy*. IEEE, 2002, Conference Proceedings, pp. 202–215.
[50] P. Ning, Y. Cui, and D. S. Reeves, "Analyzing intensive intrusion alerts via correlation," in *Recent Advances in Intrusion Detection*. Springer, 2002, Conference Proceedings, pp. 74–94.
[51] S. C. Sundaramurthy, L. Zomlot, and X. Ou, "Practical ids alert correlation in the face of dynamic threats," in *Proceedings of the International Conference on Security and Management*, 2011, Conference Proceedings.
[52] F. M. Alserhani, "Knowledge-based model to represent security information and reason about multi-stage attacks,"

in *Advanced Information Systems Engineering Workshops*. Springer, 2015, Conference Proceedings, pp. 482–494.
[53] P. Ning, Y. Cui, D. S. Reeves, and D. Xu, "Techniques and tools for analyzing intrusion alerts," *ACM Transactions on Information and System Security (TISSEC)*, vol. 7, no. 2, pp. 274–318, 2004.
[54] M. Dacier and Y. Deswarte, *Privilege graph: an extension to the typed access matrix model*. Springer, 1994, pp. 319–334.
[55] M. Dacier, Y. Deswarte, and M. Kaniche, "Models and tools for quantitative assessment of operational security," 1996.
[56] R. Ortalo, Y. Deswarte, and M. Kaniche, "Experimenting with quantitative evaluation tools for monitoring operational security," *Software Engineering, IEEE Transactions on*, vol. 25, no. 5, pp. 633–650, 1999.
[57] S. Jajodia, S. Noel, and B. OBerry, *Topological analysis of network attack vulnerability*. Springer, 2005, pp. 247–266.
[58] W. Li, R. B. Vaughn, and Y. S. Dandass, "An approach to model network exploitations using exploitation graphs," *Simulation*, vol. 82, no. 8, pp. 523–541, 2006.
[59] F. Cheng, S. Roschke, and C. Meinel, *An integrated network scanning tool for attack graph construction*. Springer, 2011, pp. 138–147.
[60] I. Chokshi, N. Ghosh, and S. K. Ghosh, "Efficient generation of exploit dependency graph by customized attack modeling technique," in *18th Annual International Conference on Advanced Computing and Communications (ADCOM)*. IEEE, 2012, Conference Proceedings, pp. 39–45.
[61] S. Noel, E. Robertson, and S. Jajodia, "Correlating intrusion events and building attack scenarios through attack graph distances," in *20th Annual Computer Security Applications Conference*. IEEE, 2004, Conference Proceedings, pp. 350–359.
[62] R. D. Shachter, "Evaluating influence diagrams," *Operations research*, vol. 34, no. 6, pp. 871–882, 1986.
[63] A. M. Agogino and A. Rege, "Ides: Influence diagram based expert system," *Mathematical modelling*, vol. 8, pp. 227–233, 1987.
[64] R. A. Howard and J. E. Matheson, "Influence diagrams," *Decision Analysis*, vol. 2, no. 3, pp. 127–143, 2005.
[65] S. Sanner, "Relational dynamic influence diagram language (rddl): Language description," *Unpublished ms. Australian National University*, 2010.
[66] R. D. Shachter, "David: Influence diagram processing system for the macintosh," *arXiv preprint arXiv:1304.3108*, 2013.
[67] M. Ekstedt and T. Sommestad, "Enterprise architecture models for cyber security analysis," in *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*. IEEE, 2009, pp. 1–6.
[68] R. Lagerström, P. Johnson, and P. Närman, "Extended influence diagram generation," in *Enterprise Interoperability II*. Springer, 2007, pp. 599–602.
[69] T. Sommestad, M. Ekstedt, and P. Johnson, "Cyber security risks assessment with bayesian defense graphs and architectural models," in *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*. IEEE, 2009, pp. 1–10.
[70] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, p. 80, 2011.
[71] S. Caltagirone, A. Pendergast, and C. Betz, "The diamond model of intrusion analysis," DTIC Document, Tech. Rep., 2013.
[72] D. L. Moody, P. Heymans, and R. Matulevičius, "Visual syntax does matter: improving the cognitive effectiveness of the i* visual notation," *Requirements Engineering*, vol. 15, no. 2, pp. 141–175, Jun 2010. [Online]. Available: https://doi.org/10.1007/s00766-010-0100-1
[73] L. Wang, C. Yao, A. Singhal, and S. Jajodia, *Interactive analysis of attack graphs using relational queries*. Springer, 2006, pp. 119–132.
[74] ISO/IEC, "Definitions and graphical notation, final draft international standard iso/iec 15909:2004," 2004. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=38225
[75] N. Ghosh and S. K. Ghosh, "A planner-based approach to generate and analyze minimal attack graph," *Applied Intelligence*, vol. 36, no. 2, pp. 369–390, 2012.
[76] J. L. Peterson, "Petri nets," *ACM Computing Surveys (CSUR)*, vol. 9, no. 3, pp. 223–252, 1977, 1755.
[77] R. Sawilla and X. Ou, *Googling attack graphs*. Citeseer, 2007.
[78] S. Nanda and N. Deo, "A highly scalable model for network attack identification and path prediction," in *SoutheastCon, 2007. Proceedings. IEEE*. IEEE, 2007, Conference Proceedings, pp. 663–668.
[79] S. Bhattacharya, S. Malhotra, and S. Ghsoh, "A scalable representation towards attack graph generation," in *Information Technology, 2008. IT 2008. 1st International Conference on*. IEEE, 2008, Conference Proceedings, pp. 1–4, cited by 4.
[80] M. Alhomidi and M. J. Reed, "Attack graphs representations," in *Computer Science and Electronic Engineering Conference (CEEC), 2012 4th*. IEEE, 2012, Conference Proceedings, pp. 83–88.
[81] S. Cheung, U. Lindqvist, and M. W. Fong, "Modeling multistep cyber attacks for scenario recognition," in *DARPA information survivability conference and exposition, 2003. Proceedings*, vol. 1. IEEE, 2003, Conference Proceedings, pp. 284–292.
[82] F.-X. Aguessy, "Évaluation dynamique de risque et calcul de réponses basés sur des modéles d'attaques bayésiens," Ph.D. dissertation, Télécom SudParis, 2016.
[83] T. Heberlein, M. Bishop, E. Ceesay, M. Danforth, C. Senthilkumar, and T. Stallard, "A taxonomy for comparing attack-graph approaches," *Online: http://netsq.com/Documents/AttackGraphPaper.pdf*, 2012, cited by 3.
[84] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis," in *Proceedings of the 1998 workshop on New security paradigms*. ACM, 1998, Conference Proceedings, pp. 71–79, cited by 523.
[85] R. Lippmann, K. Ingols, C. Scott, K. Piwowarski, K. Kratkiewicz, M. Artz, and R. Cunningham, "Validating and restoring defense in depth using attack graphs," in *Military Communications Conference, 2006. MILCOM 2006. IEEE*. IEEE, 2006, Conference Proceedings, pp. 1–10, cited by 111.
[86] J. D. Weiss, "A system security engineering process," in *Proceedings of the 14th National Computer Security Conference*, vol. 249, 1991, Conference Proceedings, pp. 572–581.
[87] E. G. Amoroso, *Fundamentals of computer security technology*. Prentice-Hall, Inc., 1994.
[88] N. B. Amor, S. Benferhat, and Z. Elouedi, "Naive bayes vs decision trees in intrusion detection systems," in *Proceedings of the 2004 ACM symposium on Applied computing*. ACM, 2004, pp. 420–424.
[89] C. Livadas, R. Walsh, D. Lapsley, and W. T. Strayer, "Using machine learning techniques to identify botnet traffic," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*. IEEE, 2006, pp. 967–974.
[90] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 649–656.
[91] C. Baker. (2011) Continuous improvement through effective root cause & corrective action. [Online]. Available: http://ruiz-torres.uprrp.edu/geop4316/

files/Root_Cause_Corrective%20Action.pdf
[92] ECSS, "Fault tree analysis - adoption notice ecss / iec 61025," 1997.
[93] BSI, "Bs en 61025:2007 - fault tree analysis," 2007.
[94] W. Vesely, M. Stamatelatos, J. Dugan, J. Fragola, J. Minarick III, and J. Railsback, "Fault tree handbook with aerospace applications version 1.1," *NASA Office of Safety and Mission Assurance, NASA HQ*, 2002.
[95] M. Stamatelatos and J. Caraballo, *Fault tree handbook with aerospace applications*. Office of safety and mission assurance NASA headquarters, 2002.
[96] C.-Y. Cheng, S.-F. Li, S.-J. Chu, C.-Y. Yeh, and R. J. Simmons, "Application of fault tree analysis to assess inventory risk: a practical case from aerospace manufacturing," *International Journal of Production Research*, vol. 51, no. 21, pp. 6499–6514, 2013.
[97] A. J. Kornecki and M. Liu, "Fault tree analysis for safety/security verification in aviation software," *Electronics*, vol. 2, no. 1, pp. 41–56, 2013.
[98] Y. E. Senol, Y. V. Aydogdu, B. Sahin, and I. Kilic, "Fault tree analysis of chemical cargo contamination by using fuzzy approach," *Expert Systems with Applications*, vol. 42, no. 12, pp. 5232–5244, 2015.
[99] H. Lambert, "Use of fault tree analysis for automotive reliability and safety analysis," Lawrence Livermore National Lab., CA (US), Report, 2003.
[100] F. Campean and E. Henshall, "A function failure approach to fault tree analysis for automotive systems," SAE Technical Paper, Report 0148-7191, 2008.
[101] J. B. Dugan, S. J. Bavuso, and M. A. Boyd, "Dynamic fault-tree models for fault-tolerant computer systems," *Reliability, IEEE Transactions on*, vol. 41, no. 3, pp. 363–377, 1992.
[102] R. Manian, J. B. Dugan, D. Coppit, and K. J. Sullivan, "Combining various solution techniques for dynamic fault tree analysis of computer systems," in *High-Assurance Systems Engineering Symposium, 1998. Proceedings. Third IEEE International*. IEEE, 1998, Conference Proceedings, pp. 21–28.
[103] R. A. Sahner, K. Trivedi, and A. Puliafito, *Performance and reliability analysis of computer systems: an example-based approach using the SHARPE software package*. Springer Science & Business Media, 2012.
[104] M. Masera, I. N. Fovino, and A. De Cian, "Integrating cyber attacks within fault trees," *Reliability Engineering & System Safety*, vol. 94, no. 9, pp. 1394–1402, 2009.
[105] P. A. Khand, "System level security modeling using attack trees," in *Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on*. IEEE, 2009, Conference Proceedings, pp. 1–6.
[106] N. Falliere, L. O. Murchu, and E. Chien, "W32. stuxnet dossier," Symantec, Report, 2011.
[107] A. Matrosov, E. Rodionov, D. Harley, and J. Malcho, "Stuxnet under the microscope," *ESET LLC (September 2010)*, 2010.
[108] S. Jha, O. Sheyner, and J. Wing, "Two formal analyses of attack graphs," in *15th IEEE Computer Security Foundations Workshop, 2002. Proceedings*. IEEE, 2002, Conference Proceedings, pp. 49–63.
[109] B. S. Bloom, *Handbook on formative and summative evaluation of student learning*. McGraw-Hill, 1971.
[110] Bloomstaxonomy.org, "Bloom's taxonomy," 2017. [Online]. Available: http://www.bloomstaxonomy.org/Blooms%20Taxonomy%20questions.pdf
[111] Qualtrics LLC, "qualtrics," 2017. [Online]. Available: https://www.qualtrics.com/homepage/
[112] C. Valasek and C. Miller, "Remote exploitation of an unaltered passenger vehicle," IOActive, Report, 2015. [Online]. Available: http://www.ioactive.com/pdfs/IOActive_Remote_Car_Hacking.pdf
[113] Boston University, "Sony hack: A nauseating whodunit," 2015. [Online]. Available: https://www.cs.bu.edu/~goldbe/teaching/HW55815/presos/sonyhack.pdf
[114] P. Elkind, "Inside the hack of the century," 14th September, 2015 2015. [Online]. Available: http://nics.syr.edu/wp-content/uploads/2016/04/Inside-the-Hack-of-the-Century.pdf

Author Biographies

Harjinder Singh Lallie holds an MSc in Information Networks, an MPhil (University of Birmingham) and a BSc in Industrial Information Technology. Harjinder is an experienced academic with more than 20 years of teaching and tutoring experience in the areas of cyber-security, computer networks and digital forensics.

Harjinder is the course leader for the MSc Cyber Security and Management degree. He teaches Digital Forensics at the University of Warwick and the University of Oxford (Centre for Doctoral Training in Cyber Security).

Harjinder has published in numerous peer-reviewed journals and conference and has sat on numerous research programme committees. Harjinder's research interests include threat modelling, digital forensics and the presentation of digital evidence.

Kurt Debattista is an Associate Professor at the University of Warwick. His research interests include high-fidelity rendering, perceptual imaging, high dynamic range imaging, and high-performance computing. Debattista has a PhD in Computer Science from the University of Bristol, an MSc in Computer Science and an MSc in Psychology.

Jay Bal holds a PhD from the University of Warwick (WMG), an MSc in Integrated IC Systems Design (UMIST), and a BSc in Electronics and Management.

Jay has worked with GEC Telecommunications as a electronic circuit techniques designer, GEC Private Systems, Plessey Research and Rover Advanced Technology Centre.

Jay's research interests include: virtual business ecosystems, virtual organisation breeding environments, creating value through the internet, electronic markets and improving the success rate of IT projects