Regression Nodes: Extending attack trees with data from social sciences

Jan-Willem H. Bullée*, Lorena Montoya*, Wolter Pieters*‡, Marianne Junger† and Pieter H. Hartel*
* Services, Cyber-security and Safety group (SCS), Faculty of Electrical Engineering, Mathematics and Computer Science
University of Twente., Enschede, The Netherlands
† Industrial Engineering and Business Information Systems (IEBIS), Faculty of Management and Governance
University of Twente., Enschede, The Netherlands
‡ ICT section, Faculty of Technology, Policy and Management
Delft University of Technology, Delft, The Netherlands TU-DELFT

***Abstract*—In the field of security, attack trees are often used to assess security vulnerabilities probabilistically in relation to multi-step attacks. The nodes are usually connected via AND-gates, where all children must be executed, or via OR-gates, where only one action is necessary for the attack step to succeed. This logic, however, is not suitable for including human interaction such as that of social engineering, because the attacker may combine different persuasion principles to different degrees, with different associated success probabilities. Experimental results in this domain are typically represented by regression equations rather than logical gates. This paper therefore proposes an extension to attack trees involving a regression-node, illustrated by data obtained from a social engineering experiment. By allowing the annotation of leaf nodes with experimental data from social science, the regression-node enables the development of integrated socio-technical security models.**

I. Introduction

The complexity of attacks on critical systems increases with the complexity of the systems themselves [1]. To evaluate the safety of systems, fault trees were first developed in the 1960s [2]. Attack trees were popularized by Bruce Schneier [3] and constitute similar tree structures which have been used since the 1990s to assess security. The root node of an attack tree depicts the goal of the attacker (e.g. Obtain Exam). The children of a node in the tree are refinements of the node's goal into sub-goals. The leaves of the tree represent the basic actions to be executed by the attacker. Relations between siblings can either be: (i) AND-relations for which all sub-goals have to be executed to satisfy the parent node; or (ii) OR-relations for which any of the sub-goals has to be executed to satisfy the parent node (refer to Figure 1 for an example of an attack tree).

The quantification of attacks is a key component in security risk evaluation and mitigation. The leaf nodes of attack trees can be annotated with quantitative information. Values such as probability of success, costs or frequency of occurrence can be estimated for each leaf node and propagated up to the root node. However, the mathematical operations corresponding to the AND and OR relations differ. In the case of an AND gate, the probability of success of an attack is represented by the product of the associated leaf nodes whilst for OR gates the MAX-function is applied to the leaf nodes. In fact, other propagation rules exist as well. For the purpose of this paper, there is no strict need to use a particular set of propagation rules.

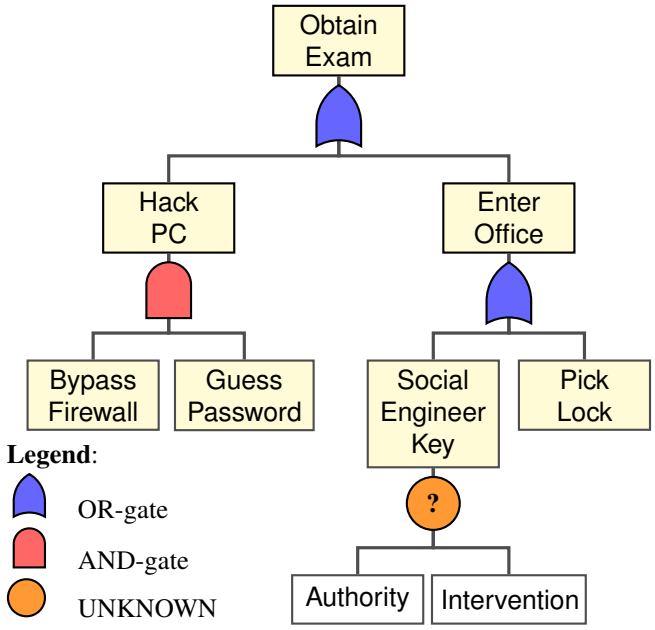

Fig. 1. An attack tree for Obtaining the Exam

The annotation of leaf nodes is usually done using expert knowledge. Typical annotations used are dichotomies (e.g. yes-no) or ordered categories such as 3 or 5-point Likert scales (e.g. low-medium-high). Although such methods suit experts because of their elegance, they have some drawbacks. In the case of ordered categories, reliability can be hampered due to biases such as the optimism bias [4], anchoring bias [5] or the overconfidence effect [6]. Another factor to be considered is

that the parent-and-child relation being modelled is context-dependent. This situation is typical of data from the social sciences involving human behaviour, such as that of social engineering experiments, in which the probability of success depends on the attacker's changes to the context by e.g. applying persuasion principles [7], which aim to maximize the probability of success.

To illustrate the modelling difficulties, a scenario consisting of an attacker who wants to steal an exam is used (refer to Figure 1). In the scenario we assume that the exam is in the office of the lecturer; physically in a printed form and digitally on the PC. The office is on the second floor of the building and can only be accessed by the lecturer who has the key. Furthermore, the PC is connected to the internet and is protected by a password that is only known to the lecturer. Finally, when the lecturer is not in the office, the office is locked. The attacker can obtain the exam by either: (i) hacking into the PC of the lecturer OR (ii) obtaining a physical copy from the office of the lecturer. To hack into the PC, the attacker needs: (a) to bypass the firewall AND (b) then guess the password of the lecturer. The key, on the other hand, can be obtained by: (a) manipulating the lecturer (social engineering) in order to obtain access to the office OR by (b) picking the lock of the lecturer's office. The success regarding the manipulation of the lecturer by means of social engineering can be influenced by: (a) the attacker using authority; OR (b) the target having received a preventive intervention; OR (c) by using both authority and intervention; OR (d) with neither the use of authority nor intervention.

The example describes both properties in control of the attacker (e.g. Bypass Firewall, Authority and Pick Lock) and properties that are in control of the target (i.e. Intervention). Therefore, the example can be interpreted as an Attack Defence Tree (ADTree) [1]. ADTrees and traditional attack trees have a comparable structure. However, there is a difference in the children; independent of the AND and OR-gates in ADTrees. Actions in ADTrees can have refinements (children of the same type) and countermeasures (children of a different type). For each refinement, there can be one sibling of a different type that counteracts the parent. In the ADTree formalism, Authority would be modelled as an attack node, whereas Intervention as a defence node.

The challenge involves modelling the node which contains human interaction (i.e. the 'Social Engineer Key' node). To manipulate the context of the attack, the attacker may use his knowledge of one or several of Cialdini's persuasion principles: authority, commitment, liking, conformity, reciprocity and scarcity (for a detailed discussion, refer to [7]). The 'authority' principle describes the likelihood of obeying requests from authoritative figures (e.g. requests made by a boss or person with a well-defined and known task). On the other hand, the potential targets can make changes to the context as well. They can be educated to protect themselves against social engineering attacks. The 'intervention', refers to an awareness campaign that helped preventing targets against social engineering attacks (e.g. describing how such an attack looks like and how to prevent becoming a victim). Such manipulations of context constitute attack steps themselves. However, this does not represent the simple type of relation which is modelled in traditional attack trees using AND or OR relations. In existing attack tree formalisms, a refinement of the node 'Social Engineer Key' could have, for example, the children: 'deploy intervention', 'get aware of the risk', 'go to office', 'increase authority' and 'request key'. In the ADTree formalism, the children 'go to office', 'increase authority' and 'request key' would be modelled as a refinement, whereas the children 'get aware of the risk' and 'deploy intervention' would be modelled as counter measures. However, the node ('Social Engineer Key') would have to be designed as either an AND or an OR node. In the former case, all 'children' would have to be executed in order to satisfy the goal of the 'parent' node (i.e. social engineer key), whilst in the latter case, the attacker would need to execute only one of the 'children'.

Such AND or OR form of reasoning does not apply to social engineering since it would be possible to succeed only by requesting the key (i.e. applying none of the 'children') but also by requesting the key and executing one or more of the 'children'. The particular nature of social engineering data (as opposed to technical data) implies that all 'children' correlate with the 'parent' to some extent; therefore, modifying any of the 'child' nodes affects the outcome (i.e. probability of success). Therefore, in an attack tree's social engineering node, combinations of 'child' nodes should yield various probabilities of success. This also means that it is possible to have a probability of compliance with the request when none of the context variables are used. A different approach is therefore needed to incorporate social engineeringhuman behaviour into attack trees. We therefore propose a 'regression node' to model the 'parent' on the basis of the 'children' parameters based on correlation coefficients as a method for incorporating social engineeringhuman behaviour nodes in attack trees.

Benefits of this method are: (i) the capability to incorporate context variables that influence probabilities, (ii) data from experiments can be used to annotate the tree and (iii) the context variables are incorporated in an compact way.

This paper is structured as follows. First, a brief overview of the proposed attack trees extension(s) is given in Section II. Section III describes the proposed regression node for attack trees. Furthermore, an example to illustrate the use of the proposed regression node is presented in Section III-A. Finally, conclusions are drawn and suggestions for future research are made in Section IV.

II. Related Work

Scholars already have made progress in the extension of attack trees. For example, multiple gates, relations and nodes have been proposed for various specific problems [8]. The extensions are separated in two groups: extensions that could model our scenario, and extensions that are interesting for future research, in particular defence nodes. For each extension method a short description is given.

A. Possible alternatives

(i) Ordered AND-gate satisfies the parent node if all children are executed according to a given order; the children can be both leaf and non-leaf nodes. The Priority AND [9], [10] and the Sequence AND [11], [12] are similar types of gates and can be used in IDS (Intrusion Detection Systems) to detect attacks that are in the execution phase but that have not yet been completed [13].
(ii) A Conditional Subordination (CSUB) gate is as extension of an AND-gate [10]. It acts as an AND-gate, with an additional side input that is prioritized over the children. The parent node is satisfied if: *a*) all children are executed or *b*) an additional side leaf which has higher priority is executed.
(iii) The Time Based Order Connector satisfies the parent node when the child nodes are executed within a predefined time frame [14]. The number of child nodes executed is at least one and the order of execution is of no importance.
(iv) The Inhibit gate is a special case of an AND-gate within the Fault Trees for Security formalism [9]. The parent's goal is satisfied if: (*a*) all the child nodes are executed; and (*b*) a predefined condition is met. The condition has to be of an environmental nature, such as temperature.
(v) XOR (Exclusive OR) gate, which originates in the Fault Trees for Security, indicates that exactly one of the children must be executed to satisfy the parent's goal [9]. This definition differs from some of the definitions of the 'standard' OR gate (i.e. at least one of the children must be executed). In the literature, these two definitions are used interchangeably. OR gates are defined as 'any child' by [3], [11], [15], 'only one child' by [16], [12], [17], [18] and 'at least 1 child' by [1], [9], [13], [19]. It should be noted that the difference in definition affects the propagation rules.
(vi) OWA (Ordered Weighted Averaging) operators are part of OWA trees [17]. The aim of OWA is to handle fuzzy sets of executed child nodes in order to satisfy the parent's goal. This implies situations that lie between 'all children' and 'one child', such as 'most of the children' or 'at least half of the children' must be executed to satisfy the parent's goal. This means that OWA allows the modelling of situations with probabilistic uncertainty based on the number of children that must be executed to satisfy the parent's goal.
(vii) In a k-out-of-n gate, the parent's goal is satisfied if a predefined subset (k) of all children (n) is executed, whereas the order is not important [10], [18]. A common way to present this is as a $\frac{k}{n}$-gate, where $k \geq 1$. In the case of $k = 1$, the function is the same as an OR-gate. A similar gate is the Threshold Based Connector, where every combination of exactly k children out of n is possible [14].

None of the extensions were able to model satisfying the parent node when none of the children are executed. Extensions (v, vi and vii) assume a minimum of 1 child is executed in order to satisfy the parent node, whereas extensions (i, ii, iii and iv) assume that all children are executed in order to satisfy the parent. It is therefore not possible to model all the 'possible' changes in context of the 'Social Engineer Key' attack step. Our approach differs because attack trees get enriched by placing basic attack actions (leaf nodes) in a particular context. The context of the attack can change. The focus is not on weighting actions *per se*, but rather on how attacks change by actions one is able to control.

B. Defence nodes

Next to possible alternatives for modelling social engineering in attack trees, defence nodes may be combined with social engineering aspects to model the effect of interventions such as awareness campaigns. ADTrees are already discussed in the Introduction. Related to this are Attack Responses, these are similar to the countermeasures in an ADTree, but they approach the problem from a different perspective [15]. Attack trees are based on all possible attack scenarios that are able to satisfy the goal of the attacker. However, Attack Responses are based on the attack consequences, (e.g. a SQL crash). The goal of Attack Responses is to find attack consequences that lead to the violation of an asset's security properties therefore, knowing all possible attack scenarios is not necessary. The Countermeasure gate can not be used to model our scenario for the same reason that the Attack Responses can not be used.

The final possible alternative is the Bayesian Belief Network (BBN). This is an approach that uses a graphical representation of prior probability distributions, represented in a Directed Acyclic Graph (DAG) [20]. Each directed edge represents a dependence relation between 2 variables, meaning that the variable (B) is stochastically dependent on variable (A), written as $P(B \mid A)$. Each node in the graph includes a table containing conditional probabilities quantifying the influence strength of the other variables [20]. Since the BBN approach is 'further' away from the approach involving the design of a new kind of node or gate, we chose not to follow the BBN option.

III. The Regression node

In traditional attack trees, all 'children' (AND-gate) or any 'child' (OR-gate) must be executed to satisfy the parent node. In this paper, we propose a 'regression-node' to model the 'leaf node' of an attack tree on the basis of the 'contextual' parameters, based on correlation coefficients. In this paper, we a first step involving the regression node modelled as a leaf node.

Regression analysis is a technique that predicts an outcome from a model, based on the relation among input variables [21]. In the 'Social Engineer Key' node presented in Figure 1, this would translate into estimating how context variables that the attacker is able to exercise or that describe him/her affect the compliance with the request to hand over the office key.

Logistic regression is used to predict binary outcomes, whereas a continuous outcome is predicted by linear regression. This means that the outcome of logistic regression is limited to the range between 0 or 1, whereas the outcome of a linear regression is any number between $-\infty$ and $+\infty$. Since the outcome of social engineering is either complying or not complying with a request, there is a need to limit the predicted

outcome to a value that is either 0 or 1, thus the need to use the natural logarithm of the odds of the predictor variable [22, p. 79-80].

In order to run a logistic regression, the dataset must fulfill three assumptions: (i) Sufficient sample size, (ii) no multicollinearity and (iii) no outliers. The dataset should at least contain 10 events per variable (EPV), which in considered as a minimum required for running a logistic regression [23]. The VIF statistic below the cut-off value of 10 indicates absence of multicollinearity [23]. In the case of dichotomous variables this means that one value should be placed in exactly one category.

In the regression node, the regression equation will replace traditional AND and OR-gates. A single regression equation consisting of: the outcome variable (i.e. dependent variable) and predictor variables (i.e. independent variables) will be used to estimate the compliance probabilities. The outcome variable is considered the construct of measurement, in the case of social engineering this is Compliance (whether or not a target complies with the request of the offender). The predictor variables are the variables that influence the outcome variable, in the case of social engineering this could be offender using authority or the target having received an intervention.

The basic logistic regression equation is:

$$LN\left(\frac{p}{1-p}\right) = \beta_0 + [\beta_1 \cdot x] \tag{1}$$

Equation 1 can be also written as Equation 2. However, for readability purposes the format of Equation 1 is preferred.

$$P(y) = \frac{1}{1 + e^{-(\beta_0 + [\beta_1 \cdot x])}} \tag{2}$$

where:

- β_0 is the intercept (i.e. constant);
- β_1 is the coefficient of the predictor variable x to the outcome y.

A general mapping from regression equation to attack tree is as follows: (i) the outcome variable of the regression equation (y) corresponds to the annotation of the parent node (e.g. Social Engineer Key) and (ii) all regression predictor variables (e.g. x) correspond to the child nodes, whereas a combination of predictor variables corresponds to one specific child node, as shown in the Example (refer to Section III-1). Only one combination of context variables applies to the attack situation; this value (probability of success) represents the final value of the regression node and is used in the propagation towards the root node. The regression node is designed as a leaf node that does not allow further refinements. Unlike in other leaf nodes, the children in the regression node do not constitute atomic actions, instead they resemble context and are used in the calculations to adapt the outcome to specific situations.

The use of the regression node will be illustrated by means of a data set from a social engineering experiment.

1) Example: Social Engineer Office Key: The dataset that is used originates from an experiment, where the objective for the attacker was to social engineer university personnel and obtain their office key. The dataset contained 3 variables: Authority, Intervention and Compliance. The 'predictor' variable Authority measured the level of formality of the attacker's clothing (e.g. jeans, t-shirt) or formal clothing (e.g. suit, tie). The variable was coded as 0 = informally dressed and 1 = formally dressed. The 'predictor' variable Intervention measured whether the potential targets have been exposed to a social engineering awareness campaign. The variable was coded as 0 = did not receive an intervention and 1 = received an intervention. The outcome variable Compliance measured whether the subject complied with the request of the attacker to hand over the office key, coded as 0 = did not comply and 1 = did comply. For more details regarding the experiment refer to [24].

The probabilities of compliance are modelled based on the equation:

$$LN\left(\frac{p}{1-p}\right) = \beta_0 + [\beta_1 \cdot x] + [\beta_2 \cdot z] \tag{3}$$

where:

- β_0 is the intercept (i.e. constant);
- β_1 is the coefficient of the predictor variable x (i.e. Authority) to the outcome y (i.e. Compliance), when z (i.e. Intervention) = 0 and;
- β_2 is the coefficient of the predictor variable z (i.e. Intervention) to the outcome y (i.e. Compliance), when x (i.e. Authority) = 0.

This equation can be easily extended to include extra predictor variables. The mapping from regression equation to attack tree is done in the following way: (i) The outcome variable of the regression equation (y) corresponds to the parent node 'Social Engineer Key' (refer to Figure 1); (ii) The children of the node 'Social Engineer Key' correspond to the predictor variables. Depending on the context that applies to the attack, that value is the 'final' value of the regression node and propagates upwards.

A. Example

Regression analyses aim at making a model based on a given dataset. To illustrate the procedure and interpret the results of the regression node the dataset from a real social engineering experiment in a university environment is used [24].

1) Example: Social Engineer Office Key: Bullée et al. explored the extent to which (i) an intervention reduces the effects of social engineering (e.g., the obtaining of access via persuasion) in an office environment and (ii) the effect of authority is of influence during such an attack [24]. In total, the offices of N = 118 employees were visited by thirty-one different 'attackers' who asked each employee (on the basis of a script) to hand over their office key. Authority, one of the six principles of persuasion, was used by half of the attackers to persuade a target to comply with his/her request. The Authority condition was operationalized by clothing:

the attacker wore either casual clothing (i.e. jeans and a t-shirt) or wore formal clothing (i.e. buttoned collar shirt and trousers). This particular dress code was used to mimic facility management personnel. Prior to the visit, an intervention was randomly administered to half of the targets to increase their resilience against attempts by others to obtain their credentials. The Intervention contained (i) an informing leaflet about the risks of social engineering attacks, (ii) a small key chain, and (iii) a humorous poster.

Among the employees that received an intervention, 37.0% handed their keys while 62.5% of those who were not exposed to it handed their key over. The intervention significantly reduced the compliance but this was not the case for authority. There was a tendency for authority to have the opposite expected effect (i.e. it works in favour of the target) [24]. Despite the authority result being counter-intuitive, the purpose is to illustrate the use of the regression-node.

The dataset (obtained from [24]) fulfilled all three assumptions needed to run a logistic regression: (i) There are at least 23 events per variable which is more than the required minimum of 10, (ii) the VIF statistic for both authority and intervention is 1.002 which is below the cut-off value of 10, indicating absence of multicollinearity and (iii) since there are only dichotomous variables used, thus dataset is free of outliers.

The outcome of the logistic regression analysis is shown in Table I. Of interest are the β coefficients, which are input for the regression equation (refer to Equation 3). The outcome of the equation are the probabilities of success for the attack tree's action 'Social Engineer Key', and range between 35% and 64% (refer to Table II).

TABLE I
OUTPUT REGRESSION ANALYSIS FOR SOCIAL ENGINEER KEY WITH DICHOTOMOUS PREDICTOR VARIABLES, N=118

	β	SE	p-value	95% CI
Authority	-.128	.382	0.739	(-.876 – .621)
Intervention	-1.051	.391	0.007	(-1.818 – -.284)
Constant	.580	.322	0.071	(-.050 – 1.211)

Although the aspects of cost/benefit are not incorporated in the regression node, one can assume that the attacker would make the rational choice to maximize the probability of success. On the basis of the Rational Choice Theory, which is used to understand human behaviour, it is assumed that: (i) the offender is a rational actor, (ii) the offender makes an end/means or cost/benefit calculation and (iii) the offender chooses to perform the behaviour based on rational calculations [25], [26]. Even if the goal of the attacker is irrational, the methods and choices to achieve it are rational [25], [27]. Therefore if the attacker manipulates the context by choosing not to apply Authority in the attack, the probability of succeeding is either 38% or 64%. Assuming that the attacker is lucky and that the target did not receive an intervention, the probability of Social Engineering the Key successfully the will be 64%.

Using a logistic regression has additional benefits regarding the representation in the tree. Alternatively, this would be an OR-gate with the Cartesian product of the context variables, resulting in an explosion of child nodes, refer to Figure 2. In the hypothetical case where there are 3 categorical variables with 3 options each, this would result in $3 \times 3 \times 3 = 27$ child nodes.

TABLE II
PROBABILITIES OF SUCCESS FOR SOCIAL ENGINEER KEY WITH DICHOTOMOUS PREDICTOR VARIABLES

Auth	Interv	Regression Equation	%Success
0	0	$LN(\frac{p}{1-p}) = .58 + [-.128 \cdot 0] + [-1.051 \cdot 0]$	64
0	1	$LN(\frac{p}{1-p}) = .58 + [-.128 \cdot 0] + [-1.051 \cdot 1]$	38
1	0	$LN(\frac{p}{1-p}) = .58 + [-.128 \cdot 1] + [-1.051 \cdot 0]$	61
1	1	$LN(\frac{p}{1-p}) = .58 + [-.128 \cdot 1] + [-1.051 \cdot 1]$	35

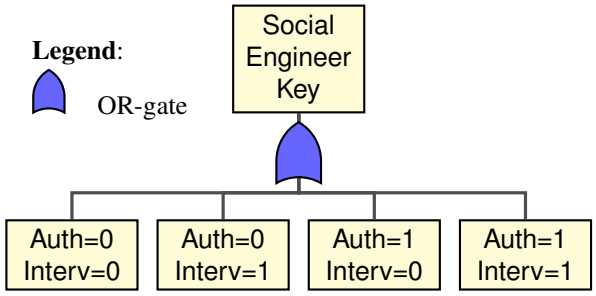

Fig. 2. An (exploded) attack tree for Social Engineer Key

B. *Further Propagation*

This section provides an example of how the Regression node result can be used to propagate to the root node. For the annotation of the other nodes in the scenario (refer to the attack tree in Figure 1) expert knowledge is used. Here we assume: (i) 80% chance to bypass the firewall and (ii) 60% chance to guess the password. Since 'Hack PC' is connected with an AND-gate to its 'children', the probability of success is calculated as the product: 80% $\times$ 60% = 48%. The probability succeeding to pick the lock is estimated at 50%.

By applying the OR-gate to Enter Office, the final result of this gate is 64%, obtained by the MAX-function of 64% (from the Social Engineer Key) and 50% (Pick Lock). Due to the higher probability of success, one would assume that the attacker would choose the option Social Engineer Key where Authority is not executed and (hopefully) the target got no Intervention, rather than choosing to pick the lock. The leaf nodes are subsequently annotated with probabilities of success (refer to Figure 3).

This regression node enables: (i) to annotate leaf nodes in the tree with context variables, (ii) incorporate data from a social science experiment in the tree, (iii) limit the explosion of terms.

IV. DISCUSSION

This paper contributes to the field of socio-technical vulnerability assessment with an attack trees extension which

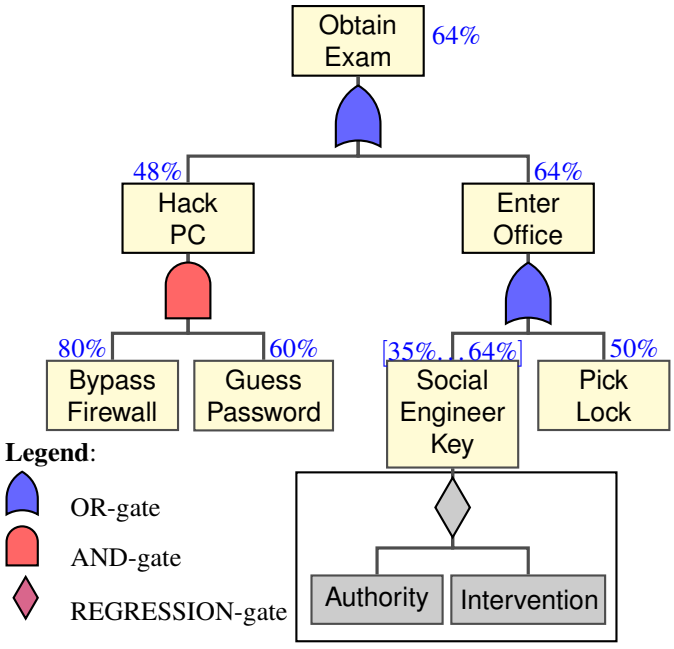

Fig. 3. A basic attack tree for Obtaining an Exam annotated with probabilities of success. The box shows the inner working of the Social Engineering node

allows the use of social science data such as that of social engineering experiments. This approach enables annotating the leaf nodes of an attack tree with success probabilities for context dependent properties which can either involve none, one, multiple or all children. The process from experimental observations towards probability of success for the root of the attack tree was illustrated using a dataset obtained from a social engineering experiment. The proposed regression node is demonstrated to work in the context of a Social Engineering experiment.

One of the main advantages of this approach is that from a graphics point of view, it can handle the explosion of terms resulting from having to display all possible combinations of context variables. Moreover, using a conventional attack tree depiction, it would not be possible to deal with continuous predictor variables since there is an infinite number of possible combinations.

The proposed extension has three limitations: (i) in its current state, the regression node constitutes a leaf node. Its output is a probability of success that can be propagated upwards using propagation rules, (ii) each experiment applied to the leaf nodes should be independent of other experiments. Therefore, since in many cases experimental data is population specific, generalization across the attack tree is not possible. This would rule out designing a single experiment to develop success probabilities for several nodes of the attack tree and, (iii) it is possible that some social science experiments are deemed unsuitable, not from a research design but from a regression assumption viewpoint since not meeting data assumptions means that the results will be unreliable and hence useless.

Finally, we make recommendations for future research. So far, a regression node for dealing with social science experimental data has been proposed. However, a follow up study should assess whether the regression node, rather than being an extension of attack trees, could constitute a means to generalize attack trees. Gates could therefore represent functions of the input. In the already proposed regression node, it is already possible to represent AND-gates. When all variables are dichotomous, it is possible to make a logical AND relation between parent and child, where the parent node becomes one if–and only if–all child nodes have of value 1. In other words, this logical AND would represent an interaction coefficient. A first step towards generalizing trees to regressions is to make the regression node available as a normal node within the tree, instead of being a leaf node.

Furthermore, the integration of the costs for the attacker into the regression gate should be also explored. This is relevant since leaving the costs out of the node means that the attacker would select the highest probability of success. This is not ideal since using persuasion principles (e.g. buying a suit) have monetary and time implications. By including these variables in the regression, the analysis becomes a cost-benefit analysis. For example, the initial authority level of the attacker with respect to a particular person in the system could be 3, and that the additional attacker cost for increasing authority by 1 unit (i.e. to 4), would carry a cost of 100.

One final question that remains is whether to assign variables to the attacker or to the defender. Authority can be an example of a property of an attacker, while intervention is an property of the defender. Since these are related, the attack tree would specify how attacker and defender properties would determine the initial values of the variables.

References

[1] B. Kordy, S. Mauw, S. Radomirović, and P. Schweitzer, "Foundations of Attack-Defense trees," in *Formal Aspects of Security and Trust*, ser. Lecture Notes in Computer Science, P. Degano, S. Etalle, and J. Guttman, Eds. Springer Berlin Heidelberg, 2011, vol. 6561, pp. 80–95.

[2] W. E. Vesely, F. F. Goldberg, N. H. Roberts, and D. F. Haasl, *Fault Tree Handbook*. Washington, DC: U.S. Nuclear Regulatory Commission, 1981.

[3] B. Schneier, "Attack trees," *Dr. Dobb's journal*, vol. 24, no. 12, pp. 21–29, 1999.

[4] N. D. Weinstein, "Unrealistic optimism about future life events." *Journal of personality and social psychology*, vol. 39, no. 5, p. 806, 1980.

[5] A. Tversky and D. Kahneman, "Judgment under uncertainty: Heuristics and biases," *Science*, vol. 185, no. 4157, pp. pp. 1124–1131, 1974.

[6] G. Pallier, R. Wilkinson, V. Danthiir, S. Kleitman, G. Knezevic, L. Stankov, and R. D. Roberts, "The role of individual differences in the accuracy of confidence judgments," *The Journal of General Psychology*, vol. 129, no. 3, pp. 257–299, 2002.

[7] R. B. Cialdini and L. James, *Influence: Science and practice*. Pearson education Boston, MA, 2009, vol. 4.

[8] B. Kordy, L. Piètre-Cambacédès, and P. Schweitzer, "Dag-based attack and defense modeling: Don't miss the forest for the attack trees," *CoRR*, vol. abs/1303.7397, 2013.

[9] P. J. Brooke and R. F. Paige, "Fault trees for security system design and analysis," *Computers & Security*, vol. 22, no. 3, pp. 256 – 264, 2003.

[10] P. Khand, "System level security modeling using attack trees," in *Computer, Control and Communication, 2009. IC4 2009. 2nd International Conference on*, Feb 2009, pp. 1–6.

[11] S. Bistarelli, F. Fioravanti, and P. Peretti, "Defense trees for economic evaluation of security investments," *2012 Seventh International Conference on Availability, Reliability and Security*, vol. 0, pp. 416–423, 2006.

[12] W. Lv and W. Li, "Space based information system security risk evaluation based on improved attack trees," in *Multimedia Information Networking and Security (MINES), 2011 Third International Conference on*, Nov 2011, pp. 480–483.
[13] S. Camtepe and B. Yener, "Modeling and detection of complex attacks," in *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on*, Sept 2007, pp. 234–243.
[14] J. Wang, J. N. Whitley, R. C.-W. Phan, and D. J. Parish, "Unified parametrizable attack tree," *International Journal for Information Security Research*, vol. 1, no. 1, pp. 20–26, 2011.
[15] S. Zonouz, H. Khurana, W. Sanders, and T. Yardley, "Rre: A game-theoretic intrusion response and recovery engine," in *Dependable Systems Networks, 2009. DSN '09. IEEE/IFIP International Conference on*, June 2009, pp. 439–448.
[16] S. Mauw and M. Oostdijk, "Foundations of attack trees," in *Information Security and Cryptology - ICISC 2005*, ser. Lecture Notes in Computer Science, D. Won and S. Kim, Eds. Springer Berlin Heidelberg, 2006, vol. 3935, pp. 186–198.
[17] R. R. Yager, "Owa trees and their role in security modeling using attack trees," *Information Sciences*, vol. 176, no. 20, pp. 2933 – 2959, 2006.
[18] A. Roy, D. S. Kim, and K. S. Trivedi, "Cyber security analysis using attack countermeasure trees," in *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research*, ser. CSIIRW '10. New York, NY, USA: ACM, 2010, pp. 28:1–28:4.
[19] I. N. Fovino, M. Masera, and A. D. Cian, "Integrating cyber attacks within fault trees," *Reliability Engineering & System Safety*, vol. 94, no. 9, pp. 1394 – 1402, 2009, {ESREL} 2007, the 18th European Safety and Reliability Conference.
[20] D. Heckerman, "A tutorial on learning with bayesian networks." Microsoft Research, Technical Report MSR-TR-95-06, 1995.
[21] A. Field, J. Miles, and Z. Field, *Discovering Statistics Using R*. SAGE Publications, 2012.
[22] A. Gelman and J. Hill, *Data analysis using regression and multilevel/hierarchical models*. New York: Cambridge University Press, 2007, vol. Analytical methods for social research.
[23] P. Peduzzi, J. Concato, E. Kemper, T. R. Holford, and A. R. Feinstein, "A simulation study of the number of events per variable in logistic regression analysis," *Journal of Clinical Epidemiology*, vol. 49, no. 12, pp. 1373–1379, 1996.
[24] J. H. Bullée, L. Montoya, W. Pieters, M. Junger, and P. H. Hartel, "The persuasion and security awareness experiment: reducing the success of social engineering attacks," *Journal of Experimental Criminology*, vol. 11, no. 1, pp. 97–115, 2015.
[25] D. Cornish and R. Clarke, *The Reasoning Criminal: Rational Choice Perspectives on Offending*. Transaction Publishers, 2014.
[26] S. K. Gul, "An evaluation of the rational choice theory in criminology," *Girne American University Journal of Social and Applied Science*, vol. 4, no. 8, pp. 36–44, 2009.
[27] Z. Winstok, "Partner violence as a rational choice," in *Partner Violence*, ser. The Springer Series on Human Exceptionality. Springer New York, 2013, pp. 47–60.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318003 (TREsPASS). This publication reflects only the author's views and the Union is not liable for any use that may be made of the information contained herein.