ESM: an enhanced attack tree model for critical infrastructure

Blaž Ivanc*/, Tomaž Klobučar***

**Jozef Stefan Institute, Jamova ulica 39, 1000 Ljubljana, Slovenia*
(Tel: +386 1 477 3900; e-mail: blaz.ivanc@determinanta.si, klobucar@e5.ijs.si)
***Jozef Stefan International Postgraduate School, Jamova ulica 39, 1000 Ljubljana, Slovenia*

Abstract: Information attack modelling can have many advantages in various stages of security audit. Although security concerns are often discussed in detail after the system has been designed, the attack modelling can prove to be an important contribution to the subsequent operational security assurance during the development of the system. In the paper, we are dealing with information attacks modelling in critical infrastructure using an enhanced structural model - ESM. In critical infrastructure, the presentation of the attacks must be appropriately demonstrated: this means sufficiently detailed and including as many details about the events during the attack. By using the enhanced structural model, proposed in the paper, the information attacks can be presented in a more detailed and transparent manner, which contributes to the improvement of security analysis during the development of systems as well as the analysis of previous information attacks.

Keywords: Attack modelling, critical infrastructure, attack tree, Enhanced Structural Model.

1. INTRODUCTION

Critical infrastructure, for example the infrastructure for transmission and production of electrical energy, traffic infrastructure, or water supply infrastructure, has a key role in functioning of today's society. The critical infrastructure is defined as an asset, system or part that is essential for the maintenance of vital societal functions, health, safety, security, economic or social well-being of people, and the disruption or destruction of which would have a significant impact on the state as a result of the failure to maintain those functions (Council Directive 2008/114/EC). Critical infrastructures are characterized by the intertwining of business information and industrial control systems, such as supervisory control and data acquisition (SCADA) systems, distributed control systems or programmable logic controllers (Stouffer et al., 2011). Compromising of the industrial control systems can lead to consequences in the physical space and consequently to major material loss, loss of supply for the population, and even loss of life.

Recently, there has been a growing interest for attacks on and protection of critical infrastructure (Alcaraz and Lopez, 2012; Hurst et al., 2014; Kim, 2014). Bologna et al., for example, found that the number of vulnerabilities in the SCADA systems detected between 2010 and 2012 was twenty times higher compared to the 2005-2010 period (Kert et al., 2014). Famous incidents and complex malicious software such as Stuxnet (Falliere et al., 2011; Langner, 2011) have shown that security of the critical infrastructure is far from perfect and needs considerable improvement in terms of security measures. (Alcaraz and Zeadally, 2015) mention several vulnerabilities of the SCADA protocols, e.g. unprotected Modbus/TCP communication, lack of authentication mechanisms, security deficiencies of the Distributed Network Protocol, or limitations of the Inter Control Center Protocol. Even though the security of the critical infrastructure in cyberspace has been the subject of numerous discussions for a long time, the existing security mechanisms cannot assure a secure and reliable operation in case of external attacks. New approaches and protection mechanism are needed to prevent different types of incidents connected to the control systems: intentional targeted attacks, unintentional incidents, and unintentional internal security events (Stouffer et al., 2011). Challenges related to secure network architectures, self-healing, modelling and simulation, and trust management and privacy in critical infrastructure also need to be discussed (Alcaraz and Zeadally, 2015).

Proper security analysis during the design time is crucial for the security provision of the infrastructure. In this paper we discuss security modelling of the information attacks on critical infrastructure. We describe an enhancement of an attack tree model called Enhanced Structural Model (ESM). The model eliminates some of the weaknesses of existing attack tree-based modelling approaches and reduces the size of the model. By using the proposed model, the information attacks can be presented in a more detailed and transparent manner, which contributes to the improvement of security analysis during the development of systems as well as during the analysis of previous information attacks. At the same time, the model is a useful tool for generating scenarios of various computer network operations.

Basic characteristics of the model have been briefly presented so far in (Ivanc and Klobučar, 2014; Ivanc and Klobučar, 2015). In this paper, ESM is presented for the first time in detail needed for the full understanding of the model and its usage in the attack modelling in critical infrastructure. Also, an upgraded version of the model is used to illustrate a real case example, and detailed evaluation of the improved ESM is provided. The evaluation includes a comparison with standard AND/OR attack tree, comparison with other enhanced attack tree models, and results of the interviews with the experts from the field.

The rest of this paper is structured as follows. Section 2 provides background and related work on attack modelling, attack tree variants, and modelling of attacks on critical infrastructure. In Section 3 new Enhanced Structural Model for attack modelling is presented in detail together with its characteristics. The use of the model on a concrete Stuxnet scenario is also illustrated. The model is evaluated in Section 4. Concluding section determines future work.

2. BACKGROUND AND RELATED WORK

2.1 Attack Modelling Techniques

Attack modelling techniques serve as a security analyst's tool and support security risk analysis.

Table 1. Advantages and disadvantages of models for attack modelling.

Model	Advantages	Disadvantages
Attack tree	Modular construction of the model, speed of construction, intuitiveness.	High abstract level, difficult to display the operations adapted to the target, difficult to capture the coordinated operations.
Attack graph	Comprehensive overview of the system security.	It reveals only known vulnerabilities and soon becomes inappropriate for use in larger systems.
Petri net	Improved capture of the coordinated actions during the attack, identifying the elements in the model.	Model quickly becomes impractically large and difficult to create.
Bayesian net	Suitable for real-time security analysis.	Methodology is not tangible enough to develop a model based on a graphic display. The issue deals with the need for statistical analysis in information security.

(Chang et al., 2010) distinguish two general approaches for attack modelling: attack trees and stochastic models. (Piètre-Cambacédès and Bouissou, 2010) categorize formal graphic security analysis models as a) static or structured models (e.g. attack trees and Bayesian networks), and b) dynamic or behavioral models with a time dimension. The latter have further been divided into low-threshold models (e.g. stochastic space-state models) and high-threshold models (e.g. Petri networks and dynamic Bayesian networks). (Fovino et al., 2009) divide graph-based attack models in two groups, namely: a) Petri-network-based models; b) attack tree models. (Hong and Kim, 2012) introduce hierarchical attack representation models. In Table 1 we present advantages and disadvantages of some of those techniques.

In this paper we mainly focus on attack trees and their enhancements. An attack tree is composed of nodes that represent attacks (Schneier, 1999). The root of the tree is the attacker's main goal, the intermediate nodes represent the subgoals, and the end nodes or leaves represent actions taken by the attacker in order to achieve the main goal. Two Boolean operations, i.e. conjunction (AND) and disjunction (OR), are associated to the root and intermediate nodes to show how the child nodes should be combined to achieve the goal. The attack tree enables modular design and the division of individual parts of the tree among several analysts (Edge, 2007). It is recommended to design higher nodes together, while individual parts are designed by specialized analysts due to the required specialist knowledge, The connection of modules into a complex attack tree is then once again carried out with teamwork.

2.2 Attack tre1e enhancements

In the past years, a number of extensions to a basic attack tree model were described. (Khand, 2009) proposed five additional types of nodes that could help customize the attacks to individual systems and their characteristics. First, there are two types of AND-nodes in the tree structure: AND-nodes that do not have the presumed implementation order for the sub-nodes, and the priority AND-nodes that provide the implementation of all the descendants of the node, starting from left to right. The author also proposed the k/n node, which requires the implementation of a certain number of all the mentioned sub-nodes. In addition, the conditional subordination node and the housing node are used for the presentation of a potential threat posed by a stakeholder in the system and not by an external attacker.

(Ariss et al., 2011) discuss the problem of high level of abstraction presented by the standard attack tree model for secure software development. As the model does not reveal the behaviour of the system when a threat occurs, the authors focused on capturing the dynamic behaviour of the system and demonstrating the attacks on a lower level of abstraction. The activities are described in terms of system behaviour and illustrated with a state chart. (McDaniel and McLaughlin, 2012) state that the attack trees lack the system details and present a model composed of two different trees. The first attack tree is based on the architectural knowledge, but in a very general form, so that it can be used in any other system that corresponds to the initial list of a certain goal. The end-node in the first tree represents a root of the second attack tree, the purpose of which is to display the attacks relating to a specific system.

(Bistarelli et al., 2006) define a defense tree, i.e. attack tree, in which the end nodes are assigned security countermeasures. (Kordy et al., 2011) propose the attack-defense tree, where an individual attack tree example is presented with a standard attack tree model. The tree structure is followed by the demonstration of a new tree structure with defense or protective characteristics. The latter also includes AND/OR nodes, but has a smaller set of nodes. The attack tree and the mentioned tree with security countermeasures are not related to each other with links; however, the defense tree structure is reasonably located under the selected set of end-nodes of the attack tree. The defense tree structure can include an additional attack tree that displays the attack on the security countermeasures. Attack countermeasure trees (Roy et al., 2012) comprise three different types of nodes that distinguish between the following events: the attack event, attack detection event, and attack mitigation event.

Table 2 presents different versions of the attack tree model that include modelling of defensive measures.

Table 2. Description of the versions of the attack tree model and their characteristics.

Model	Characteristics
Defense Tree (Bistarelli, Fioravanti, Peretti, 2006)	Presentation of defence nodes
Protection Tree (Edge, 2007)	Mapping of the attack tree in a protection tree, in which the nodes with the same position have a protective role.
Attack Countermeasure Tree (Roy, Kim, Trivedi, 2012)	Distinction between the nodes that coincide with various events during the attack.
Attack-Defense Tree (Kordy, Mauw, Radomirovic and Schweitzer, 2011)	Combination of the attack and defense tree, which are mutually and reasonably covered.

2.3 Modelling of attacks on critical infrastructure

The attack modelling in critical infrastructure can present functioning of different parts of the malicious code in a transparent and comprehensive manner. Due to its characteristics, the use of the attack tree in attack modelling is practical and already used in several studies for modelling attacks on the critical infrastructure.

(Li et al., 2010) used the method in connection with energy meters. The main goal of their attack tree is an attack on a microprocessor. The attack tree contains several AND and OR nodes, and is equipped with countermeasures as well as a node representing external errors. The nodes are marked with symbols and described in a table. With the model it is possible to find the critical path of the attack.

Different authors used attack trees for analysis of the SCADA systems security. (Lopez et al., 2012), for example, showed how attack trees can be applied for assessment of security controls for the SCADA systems, while (Bobbio et al., 2013) used for security analysis weighted attack and defense trees.

(Zhao et al., 2014) used attack tree model as a base at the approach to identify malicious code on systems. Extended model allows more flexible approach to organizing and using rules. Furthermore, a combination of static and dynamic analysis enables better accuracy and execution performance.

(Mouratidis and Giorgini, 2007) presented a new approach in dealing with the scenario-based procedures to test the security system at the design time. Their approach is mainly intended for system developers to identify critical security vulnerabilities in the early stages of the development process.

The MORDA (Mission-Oriented Risk and Design Analysis) methodology was developed for evaluation of the information system designs (Buckshaw et al., 2005). The methodology is composed of different reviews of security aspects, and its goal is to set up a defence strategy. For the purpose of analysis, the methodology uses the attack tree built on the basis of data acquired during several initial processes. The results of the attack tree analysis represent the basis for risk assessment.

A supplemented attack tree was defined by (Wang et al., 2010). The authors show how it can be used for modelling distributed denial of service (DDoS) attacks and provide an algorithm for attack detection.

3. ENHANCED STRUCTURAL MODEL FOR ATTACK MODELLING

Although the attack tree enables the use of a modular approach and inclusion of several experts from various fields and also relative transparency, the frequent criticism of the above mentioned attack tree-based models refers to their highly abstract level, difficulty of presenting the attacks adapted to a certain system, and capturing the operation of several attackers. In this section, we provide a detailed description of an ESM for attack modelling that aims at eliminating some of the deficiencies of the existing models. In comparison to (Ivanc and Klobučar, 2014; Ivanc and Klobučar, 2015) where the model was first mentioned, the detailed characteristics of the model are presented here for the first time, as well as illustration of the use of the improved model on a real case scenario.

3.1 Characteristics

The enhanced model is based on the systematic exploitation of the informative value of the model structure, with the aim to reduce the abstract presentation of the attacks and adjust individual attacks to a particular system, while at the same time enable more dynamic distribution of the attack sequence. The security countermeasures can be found in the ESM above the individual sub-tree structures of the model, since the aim of the attack described by the sub-tree structure is to neutralize security countermeasures on the way to achieving the goal of the attack. The main differences of ESM in comparison to the basic attack trees are the following:

- additional nodes,

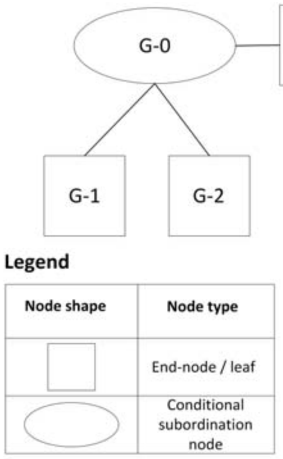
- integrated information on exploited vulnerabilities,
- integrated information on attack vectors,
- integrated information on countermeasures,
- segmentation of the attack tree.

3.1.1 Additional nodes to demonstrate the implementation of the attacks

Besides the standard set of AND- and OR- nodes ESM includes two additional types of nodes, both proposed by Khand: a conditional subordination node and a housing node. A conditional subordination node enables taking into account internal enemy as a threat agent during the attack. Use of the housing node allows us to demonstrate different time stages of the critical infrastructure operation or different process applications of industrial computers subject to internal security policy.

Conditional subordination node:

Figure 1 shows the conditional subordination node with its goal G-0. The goal G-0 is achieved if the goal in the initiator node labelled as P-1 has been achieved; or if all goals of sub-nodes – descendants, in this case G-1 and G-2, have been achieved. The initiator node can be presented as an end-node or any larger independent tree structure.

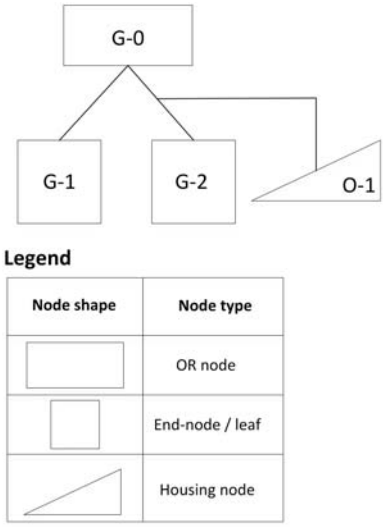


Node shape	Node type
□	End-node / leaf
○	Conditional subordination node

Fig. 1. An example of the conditional subordination node with its main goal G-0.

Housing node:

Figure 2 presents a housing node suitable for the presentation of attacks on time-varying states and different threat agents. Housing node refers to the connection between the intermediate node and the descendant node which is impacted by the housing node. In the implementation of the goal in Figure 2, the housing node can be "turned off", which means that achieving of the G-0 goal of "malicious code spreading", which is an OR-node, was performed through removable drives or through the network. In case the housing node is "turned on", this indicates that the goal of the G-0 node has been achieved either with the help of the G-1 node or the housing node, which provides for "spreading through project files". This indicates that the spreading of malicious code did not occur inside the local network.

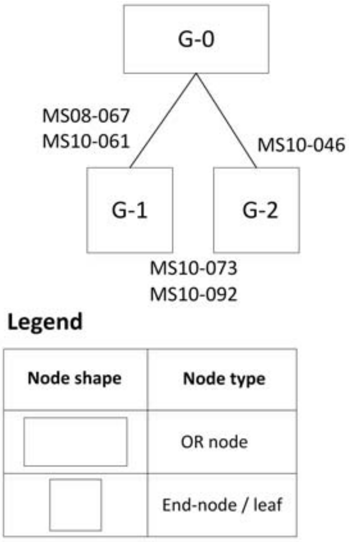


Node shape	Node type
▭	OR node
□	End-node / leaf
◢	Housing node

Fig. 2. Example of a housing node marked with O-1.

3.1.2 Labels for exploiting vulnerability

To identify more precisely the software or system under attack, we use vulnerability labels. The labels can also tell us information complexity of an attack and allow identifying the stage of the attack. Since the vulnerability labels already express the properties of the system, they are located on the lower levels of the model or at the end-node level. When dealing with vulnerability labels, it should be considered that the labels for malicious code spreading are placed above the node of interest, while the vulnerability labels for elevation of privileges are placed below the node of interest. This is demonstrated in Figure 3. Labels for exploiting vulnerability in the computer processing of the model enable linking data from different databases and public intelligence services and thus quickly inform analysts about the vulnerabilities.



Node shape	Node type
▭	OR node
□	End-node / leaf

Fig. 3. Example of vulnerability labels given above and below the end nodes.

3.1.3 Attack vector labels

In ESM, the attack vector labels are used for better explaining the techniques used during the attack. Here, it is important not to confuse attack vectors with payload. The latter is enabled with the method determined in the attack vector.

The attack vector labels must be located on the links to the end nodes, where a new vector appears in each sub-tree. An example of the label is shown in Figure 4. The vector labelled v1 is located on the link between the root of the tree and the end-node. This means that the implementation of the attack using the method, which was presented with the vector in question, is planned in this end-node. Vector labelled v2 is located between the intermediate node and the end-node, in which the use of the method presented with the vector is planned.

Each sub-tree can have several different attack vectors. The course of the attack can be selected on the basis of a minimum number of different vectors on the way to achieving the goal of the operation or select a path, in which the use of attack vectors is less critical for the attacker and enables a more probable reaching of the goal of the attack. Vector labels present an important value to security analysts in the field of risk assessment, where the threat to the system is identified through the catalogue of threats and experience, while the actual process of executing the threats is neglected or is subject to security policy revaluation.

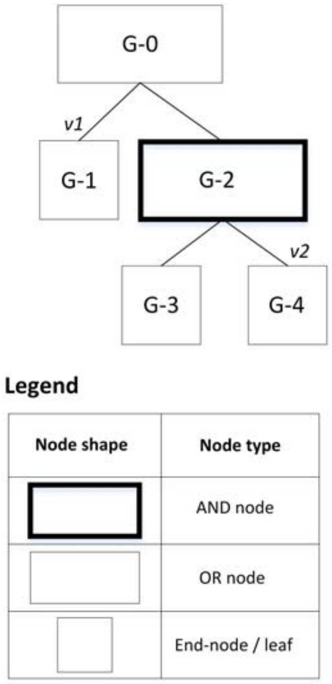


Fig. 4. Examples of attack vector labels (“v1” and “v2”).

3.1.4 Demonstration of countermeasures

Countermeasures in the ESM are represented as a set of security countermeasures. Demonstration of countermeasures provides additional data on security mechanisms and protection systems which should be considered when analyzing the attack implementation. The set is graphically represented with a node, similar to an end node. The link between the node of a set of countermeasures is then connected to the tree structure in the centre of the link between the root of the tree and its descendants (Figure 5). Several labels for the sets of countermeasures can be connected to the same link. Sets of countermeasures are labelled with the letter C and a serial number in accordance with the tree structure reading methodology.

Countermeasure labels are aimed at analysts who use them to demonstrate which sets of security countermeasures have been present in the previous information attack analysis, but due to various reasons have not partially or completely prevented the payload. In modelling the future or hypothetical information attacks, security countermeasures are first placed on the basis of the information received by the analysts. Then, attack nodes (marked with a catch-letter “G”) are used to model the attack which will bypass, nullify or in any other way neutralize the security countermeasures.

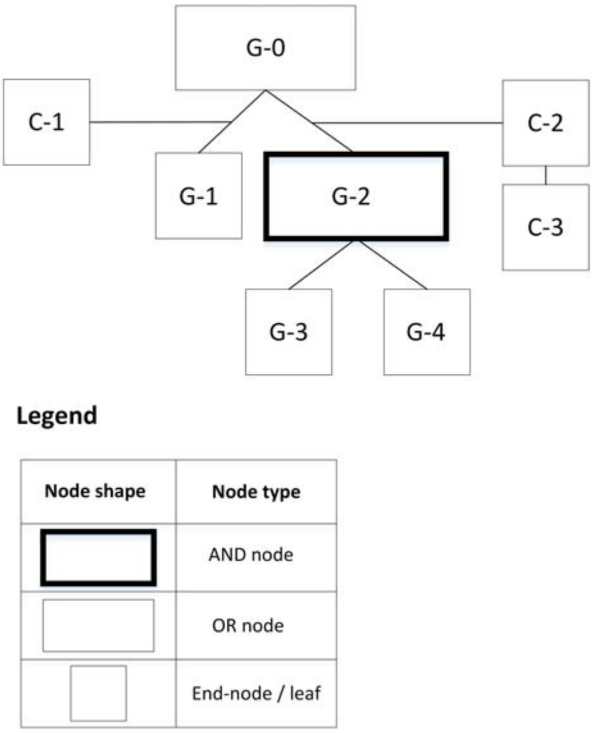


Fig. 5. Example of a countermeasure label (“C-1”, “C-2” and “C-3”).

3.1.5 Segmentation of the attack tree

Sub-trees in ESM having specific characteristics can be labelled as segments. This approach enables, for example, the analysts specialized for certain types of attack to indicate when and where the model starts to reveal system-specific attack scenarios. Separation of the tree in different segments also allows modelling work on each segment to be conducted in parallel. In the final structure of the model, the code segments can be used afterwards and thus isolate certain characteristics of the course of the attack. Isolated part is labelled with the letter S and a serial number, and separated from other parts with a dotted line (Figure 6). At the same time, a description with the meaning of each segment label is given in the relevant table. Labelling segments enables better work coordination in assembling and analysing the model and directs the need for input data, such as specific attack techniques and procedures, selected security mechanisms, and vulnerability labels.

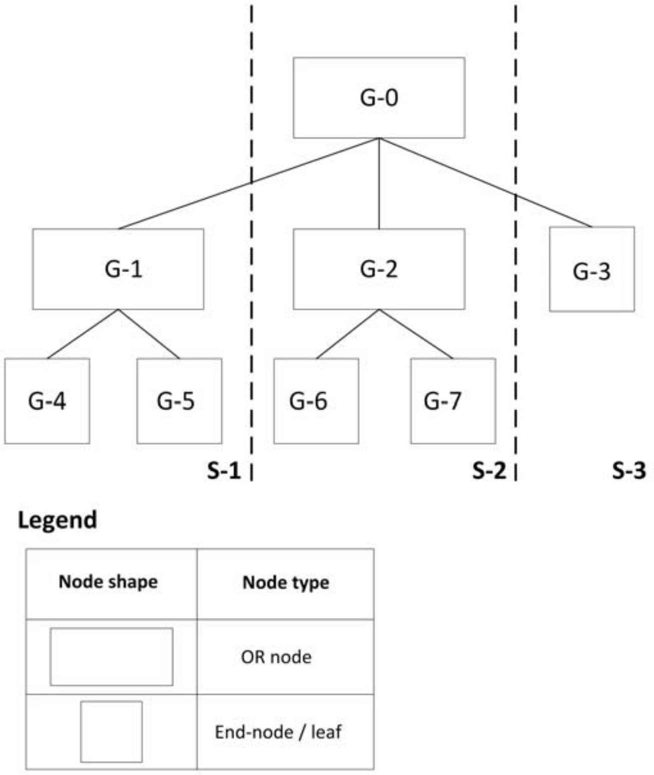


Fig. 6. Demonstration of a segment label ("S-1", "S-2" and "S-3") in the tree structure.

3.2 Illustration of an example

In this section, we illustrate usefulness of the proposed ESM on a concrete real life example of the offensive computer-network operation called Stuxnet. First, we give a short description of the attack scenario. This is followed by a demonstration and reading of ESM.

An attack scenario includes the spreading of malicious code and compromising of programmable logic controllers. The complex malicious software called Stuxnet is an example of one of the most prominent information attacks in recent years. It was discovered in June 2010 and followed by a number of analyses of the attack. The targets of the attack were the facilities for enrichment of uranium in Iran. The purpose of the attack was to reprogram the industrial control systems by changing the PLC code. The aim or the consequence of the attack resulted in changes in the physical process of the critical infrastructures. In order to perform the attack that could take place unnoticed, a high degree of completion of the attack and the ability to compromise different systems were required. The malicious code dropper exploited the unknown vulnerabilities in several versions of the Windows operating system to spread the malicious code within the target systems. The mentioned exploited vulnerabilities are presented in Table 3.

The malicious code Stuxnet has spread inside the target in several ways. In terms of the exploitation of software vulnerabilities, the code has spread within the network by exploiting two vulnerabilities and one on the level of removable media. Two additional vulnerabilities were exploited for the elevation of privileges that were required for installing the malicious code. The selection of these vulnerabilities was dependent on the host computer operating system. Infection with malicious code could have been executed through project files of the dedicated software.

Spreading through remote drives and project files enabled the transfer of the infection to isolated computer systems. The final system targets of the attack were SIMATIC WinCC control system, SIMATIC Step7 engineering system and the selected SIMATIC programmable logic controllers (6ES7-315-2, 6ES7-417). The key for the target payload was in PLC rootkit, the code of which was in .DLL file. By replacing the .DLL file, the hidden operation of the malicious code and a practical execution of the man-in-the-middle concept were enabled. The latter enabled the modification of the PLC code and concealing of the actual situation to the operators.

Table 3. Description of vulnerabilities exploited in the Stuxnet information attack.

Vulnerability label	Description
MS08-067	Windows Server Service Vulnerability
MS10-046	Windows Shell LNK Vulnerability
MS10-061	Windows Print Spooler Service Vulnerability
MS10-073	Windows Win32K Keyboard Layout Vulnerability
MS10-092	Windows Task Scheduler Vulnerability

Changes in the operation of programmable logic controllers (PLC) are achieved by replacing the .DLL file, which enables monitoring of the reading and writing of the code data blocks to or from the PLC, PLC infection by inserting its own blocks or replacements and changes in the existing blocks. At the same time, the replacement of the .DLL file also allows to hide the PLC infection. PLC infection process begins when the desired PLC model and system data blocks, which show the use and scope of the frequency converters in question, have been verified or detected (Falliere et al., 2011).

Figure 7 presents an ESM, which describes the scenario attack presented above. In addition to AND/OR nodes and end-nodes, the model includes the conditional subordination node and housing node, vulnerability labels, attack vector labels, countermeasures and segmental distribution of the model structure. The description of nodes is given in Table 4. The model is divided in several segments and the description of the latter is given in Table 5. Table 6 includes the description of the attack vectors that are labelled on some of the links between the nodes. Table 7 presents the sets of countermeasures that are demonstrated on the model.

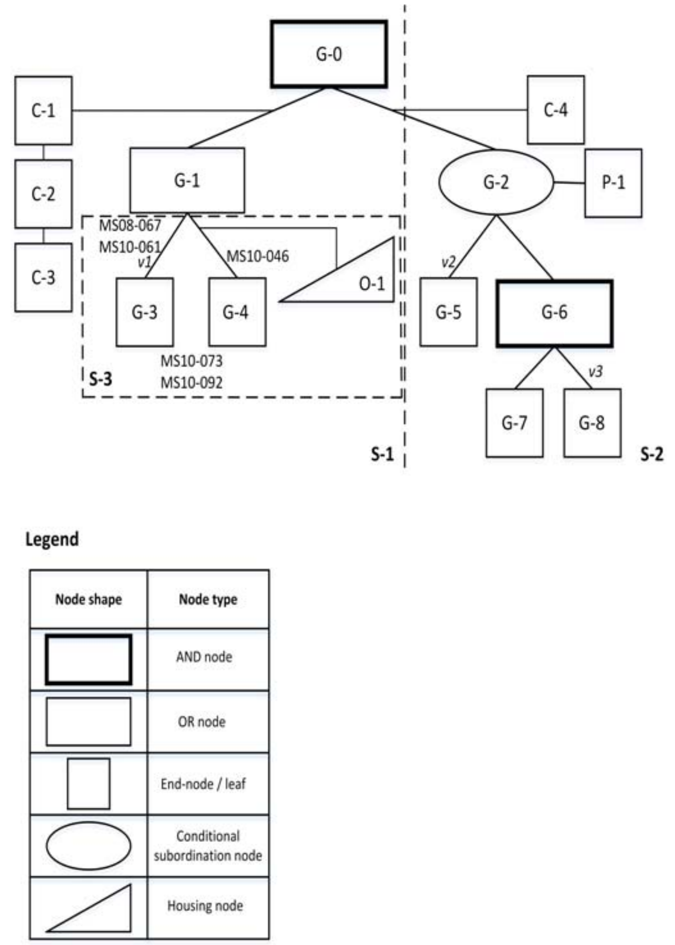


Fig. 7. ESM performed on the basis of the Stuxnet malicious code operation.

Table 4. Description of nodes for the ESM in Figure 7.

Node	Description
G-0	System sabotage in critical infrastructure
G-1	Malicious code spreading
G-2	Compromising of the industrial control system components
G-3	Spreading through the network
G-4	Spreading through removable media
G-5	Malicious replacement of the s7otbxdx.dll file
G-6	Change in the operation of target programmable logic controllers (PLCs)
G-7	Verifying SDB blocks
G-8	Changing the data sent/returned from PLC
O-1	Spreading through project files
P-1	Insider [alternative option to perform the attack]

Table 5. Description of segments labelled in the ESM in Figure 7.

Segments	Description
S-1	Dropper
S-2	Payload
S-3	Structure of malicious code spreading

Table 6. Description of the attack vectors labelled in the ESM in Figure 7.

Vector	Description
v1	Attack on weak authentication
v2	DLL hijacking
v3	Implementation of the man-in-the-middle mechanism

Table 7. Description of security countermeasures labelled in the ESM in Figure 7.

Countermeasure	Description
C-1	Network-isolated systems
C-2	Control over the events in computer resources
C-3	Network system for intrusion detection
C-4	Authentication security mechanisms

Model reading

The three segments of the model are labelled as: S-1, S-2, and S-3. S-1 presents the dropper. S-2 presents the sub-tree structure that illustrates the target payload. S-3 presents the sub-tree structure that is focused on malicious code spreading.

The main goal of the attack is the node, which is the root of the tree labelled as G-0 “system sabotage in critical infrastructure”. The node requires the implementation of both sub-nodes. Sub-node G-1 “malicious code spreading”, which is also the goal of the sub-tree structure indicated with the S-1 segment, is implemented with one of the sub-nodes: G-3 “spreading through the network” or G-4 “spreading through removable drives”. The links from nodes G-1 to G-3 and G-1 to G-4 include vulnerabilities that are exploited with the purpose of spreading the malicious code. Below the nodes G-3 and G-4, there are labels for vulnerability that are exploited for the elevation of privileges that are necessary for installing the malicious code. Housing-node O-1 presents the possibility of spreading through project files. The node is used to illustrate the attack that enabled the infection of the systems on the isolated nodes. Therefore, O-1 node excludes the use of G-4 node.

G-2 node “compromising of the industrial control system components” presents a conditional subordination node. This requires the implementation of both sub-nodes, namely nodes G-5 “malicious replacement of the file” and G-6 “change in the operation of target programmable logic controllers (PLCs)”. To achieve this goal, the implementation of nodes G-7 “verifying SDB blocks” and G-8 “changing the data sent and returned from PLC” is required. The goal of the G-2 node can also be achieved with the initiator node – P-1, which provides for the compromising of the industrial control equipment prior to the installation or by an insider.

Three attack vectors are labelled in the model. The method labelled with the attack vector v1“weak authentication” is used with the end-node G-3 “spreading through the network”. “DLL hijacking” method labelled with the attack vector v2 is located in the link to the end-node G-5 “malicious replacement of the s7otbxdx.dll file." This is important, since the rootkit software code for programmable logic controller was entirely located in a fake file s7otbxdx.dll. The attack or use of the man-in-the-middle mechanism is labelled with the vector attack v3. This mechanism allowed the modification of the data sent or received from the controller without the knowledge of the operator.

Four sets of countermeasures are labelled in the model. Due to various concrete security solutions that are classified in the same set, the elements of the latter are not listed. Countermeasure labels are placed relatively high in the model, as for example the implementation of the attack presented with a sub-tree structure and with its main goal G-1 "malicious code spreading" provides for the neutralization of security countermeasures presented with the nodes labelled as C-1 "network-isolated systems", C-2 "control over the events in computer resources", and C-3 "network system for intrusion detection". The same refers to the rest of the operation presented with a sub-tree structure and with its main goal G-2 "compromising of the industrial control system components" and requires a neutralization of security countermeasures which can be classified in a set of countermeasures presented with the C-4 node "authentication security mechanisms".

4. EVALUATION

The proposed model has been compared with a basic attack tree model and with other enhanced attack tree models, as well as evaluated by field experts. The evaluation procedure and results are described below.

4.1 Comparison with a basic attack tree model

Due to the modular design enabled by the tree structure, the comparison with the basic attack tree model representation of the same attack, depicted in Figure 8, is presented in two parts. Firstly, we deal with the sub-tree structure with the main goal G-1 "malicious code spreading". Secondly, we present the sub-tree structure with its main goal G-2 "compromising of the industrial control system components". In the ESM, these two sub-tree structures are additionally divided with segments. With the exception of the segment S-3 "structure of malicious code spreading" which is located in an ESM, the demonstration of other two segments is not required. The presentation of the models does not include the demonstration of the set of countermeasures, since they are not provided for in the standard attack tree.

Comparison of the sub-tree structure with its main goal G-1:

The most evident difference between the two models is the number of nodes and levels. The basic attack tree model of the sub-tree structure in question contains 14 nodes, while the ESM contains only four. The reason for this is the use of vulnerability labels that can be found in the ESM. This eliminates the nodes G-10 and G-14 to G-19 from the attack tree model. In the attack tree, we can notice that it contains two identical sections: these are two smaller sub-tree structures with the same goals – G-9 and G-11 – and with identical content of their nodes – descendants, which are in fact the end nodes. This is due to the fact that the tree structure is a non-cyclic graph, which in this case requires the repetition of certain parts in the lower levels of the structure.

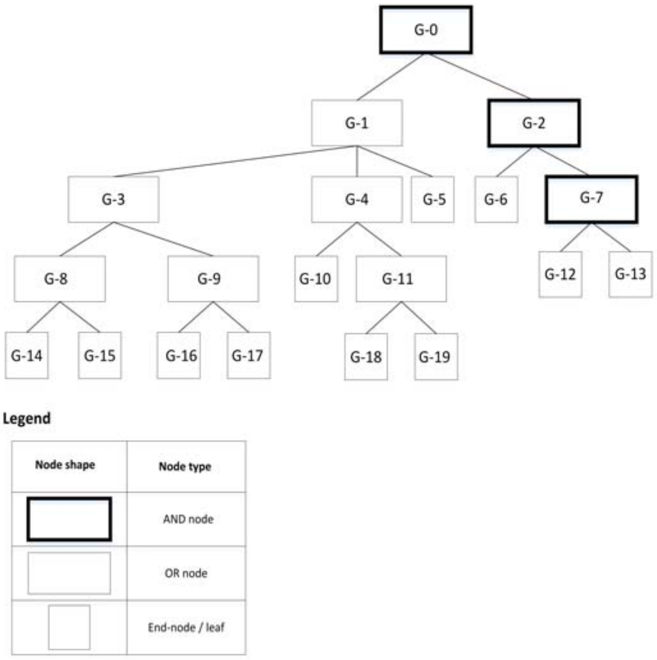


Fig. 8. Standard attack tree model with AND/OR nodes performed on the basis of the Stuxnet malicious code operation.

Table 8. Description of the nodes for the basic attack tree model in Figure 8.

Node	Description
G-0	System sabotage in critical infrastructure
G-1	Malicious code spreading
G-2	Compromising of the industrial control system components
G-3	Spreading through the network
G-4	Spreading through removable media
G-5	Spreading through project files
G-6	Malicious replacement of the s7otbxdx.dll file
G-7	Change in the operation of target programmable logic controllers (PLCs)
G-8	Spreading by exploiting vulnerability
G-9	Installing malicious program code
G-10	Exploiting MS08-046 vulnerability
G-11	Installing malicious program code
G-12	Verifying SDB blocks
G-13	Changing sent/inserted data from PLC
G-14	Exploiting MS10-061 vulnerability
G-15	Exploiting MS08-067 vulnerability
G-16	Exploiting MS10-073 vulnerability
G-17	Exploiting MS10-092 vulnerability
G-18	Exploiting MS10-073 vulnerability
G-19	Exploiting MS10-092 vulnerability

This is another advantage of the ESM that in some cases eliminates the above-stated. Therefore, the methodology for placing the vulnerability labels above the respective nodes is used to label the exploited vulnerabilities for a direct spreading of the malicious code, while the vulnerability labels below the nodes refer to the elevation of privileges required to install the malicious code.

The presented part of the ESM contains housing node O-1 that is placed on the link between the nodes G-1 and G-4. The action of the housing node in the ESM is presented independently in the attack tree with the G-5 end node. Although there is no difference at first glance, the latter is evident in the analysis of the two structures. In the attack tree, the goal of the G-1 node is achieved with one of its nodes – descendants: G-3, G-4 or G-5. The same refers to the ESM, but with one significant difference: if the goal has been reached with the action presented by the housing node O-1, the G-4 node has not been used. This clearly states that the action performed using the housing node O-1 could not previously take place in the path with G-4 node.

Comparison of the sub-tree structure with its main goal G-2:

The sub-tree structures with their main goal G-2 have certain similarities in terms of the basic attack tree model and the ESM. The ESM has one additional node. This is initiator node labelled as P-1, which does not have any corresponding node in the attack tree example. This node is the result of the fact that the G-2 node presents a conditional subordinate node in the ESM. To achieve the goal, the conditional subordinate node requires implementation of all of its nodes – descendants, while at the same time enables an alternative implementation of the goal. The attack tree, of course, includes only standard AND-node. In order to present an alternative implementation of this goal, we would have to transform the G-2 node into the OR-node. In doing so, one link would be intended for the end-node, which would present an alternative implementation. Another link would be intended for a new node, which requires another level in the tree structure for the presented situation, which requires the implementation of both sub-nodes – descendants.

The ESM includes two attack vector labels. This enables the analyst to additionally communicate the method that was used in implementing the attack in a certain part of the model. This is the extended functionality of the model, which does not impact the graphical illustration of the attack, i.e. the number of nodes and levels. In order to take into account the attack vectors in the attack tree model, the description of each node in the table would have to be expanded by listing both the action and the method.

4.2 Interview-based evaluation

For the purpose of ESM evaluation, 6 structured interviews were conducted with experts from the field of modelling and critical infrastructure in Slovenia. The interviewees from academia, armed forces, police, government and corporate security were selected according to their area of expertise and experience with threats and risks related to critical infrastructure, as well as modelling, risk evaluation and cybernetic systems.

The interview comprised of ten questions which were divided into two parts: the general part and the part pertaining to the ESM. The questions in the general part related to the interviewees' work experience and their view about the situation of critical infrastructure protection in the Republic of Slovenia. In addition, the interviewees expressed their opinion on the importance of attack modelling within the investigated field, current development of this approach and disadvantages of the modelling itself. The second part of the questionnaire referred to the ESM from the following perspectives: improvement of modelling by means of structural or static techniques, model's application limitation, model advantages and disadvantages, as well as model's contribution to the improvement of critical infrastructure protection.

The interviews were conducted individually on a face-to-face basis. In the beginning, the interviewees were given a brief presentation listing the properties of the ESM. The presentation was concluded with a short illustration of the attack tree and the ESM in the same case. The average interview was held for 45 minutes.

According to the interviewees, the most commonly addressed weaknesses in the critical infrastructure protection are lack of awareness, lack of understanding of the actual danger, unfamiliarity with the security threats, and insufficient system security design. Interviewees perceive attack modelling within critical infrastructure as highly important. However, according to them, only a handful of experts are engaged in this type of activities which are quite undeveloped in the Republic of Slovenia. Some of the reasons for underdevelopment are poor knowledge of modelling and under-qualified staff, as well as poor software tools support for modelling.

The conducted interviews show that the ESM includes explicit elimination of specific limitations and difficulties which are otherwise present in modelling by using these types of models. The ESM provides effective solutions to the security-related problems within critical infrastructure in question. The participants should have had more comprehensive knowledge of the model in order to provide a more reliable evaluation. The application level of the model would have been better reflected if illustrated with a more complex scenario.

The model still needs to become recognised within the scientific circle and calls for further software-related support. In addition, some of the interviewees clearly stated that the model can also be transferred to other environments and can be used as a valuable tool by other experts, not only by analysts engaged in attack modelling

4.3 Comparison with enhanced attack tree models

The purpose of this chapter is to discuss the advantages and disadvantages of the attack tree, and compare the proposed model with other approaches.

The limitations of the attack tree model can be found described in the papers dealing with Petri network modelling. (Chen et al., 2011) say that the attack tree model deals with attacks only through the step-by-step approach. They have focused on a single goal of the attack and a single attacker. The authors highlight that the attack tree cannot completely reveal the coordinated operation, which can be assigned to the tree structure of the model. The main goal of the model presented in the root of the tree is considered as an individual goal of the attack. At the same time, the nodes between the root of the tree and the end-nodes in the attack tree model represent partial or intermediate goals. The proposed solution to this problem is to design a set of models that would form the so-called attack forest. In addition, the ESM is suitable for construction of the attack forest. At the same time, the model with a larger set of nodes and additional information on the method of the attack strives to cover the coordinated action efficiently. The basic tree structure models are difficult to provide a quality illustration of the coordinated action.

(Dalton et al., 2006) point out that in the attack tree modelling, it is difficult to reuse or divide certain attack trees. This is attributed to the absence of standardized construction methods, simulation and analysis of the model. At the same time, they note the need to select field experts who should be involved in modelling. With the use of labelling features, the ESM allows the analysts to clearly label the individual sub-tree structure, which can be reused or updated where necessary.

(Pudar, 2009) state the difficulty of the attack tree model in providing the additional information that is available in relation to the event. Therefore, the attack tree model presents a mere action of the attack without any additional information about the event of attack that may be available. At the same time, they mention the insufficient accuracy in the presentation of the attack within the model analysis. An additional weakness is a static model, which means that the latter is valid until any changes appear in the system. By labelling the segments in the ESM, we can influence the model operators and analysts to periodically update each model in accordance with the system changes. By using all of its features, the ESM enables provision of information about the event, e.g.: exploited vulnerability in an individual action of the attack, the method used in performing the payload, and neutralization of security countermeasures. Changes in the system which require adjustment of the model can be easily applied: the emergence of new or elimination of the existing vulnerabilities can be replaced without modifying the content of an individual node or changing a certain part of the model. The same refers to the emergence of new invasive methods, which also do not require any changes in the content of the nodes or in the model structure.

In this review, we should mention certain limitations that refer to attack modelling based on Petri networks, particularly in the critical infrastructure systems. (Chen et al., 2012) mention that in this case, the model becomes impractically large and difficult in terms of designing. At the same time, the designer requires considerable expert knowledge when assembling the model. In this case, modular design, intuitive design and reading typical for models based on the tree structure are much more convenient, since they facilitate the inclusion of experts from different fields in the modelling.

It is preferable that several analysts from different fields work on a structural model, such as the attack tree. In this way, sufficient professional model can be developed. Therefore, it would be reasonable to insert segments that divide certain parts of the attack into an individual, specialized field. Certain segments can present a universal section of the tree structure model for a chosen implementation of the attack.

Using the elements from the ESM, we can compose a similar but yet much more complex attack model. The attack tree has a weak informative form, since it does not exploit the links between the nodes for placing vulnerabilities or attack vectors. For this reason, this content can be given only in the description of the nodes, which presents a greater amount of space. In addition, there is only one input field – the description of the node. Without any greater information value, it may happen that a certain sub-tree structure will have to be repeated in the same attack tree model, as shown in the previous chapter.

The proposed ESM could be used in practice as a data structure for generating scenarios in the automated exploitation frameworks. In addition, the model can be used as a software-based tool for management of computer-network operations.

5. CONCLUSION AND FUTURE WORK

In attack modelling, it is particularly important that the design of the entire attack is carried out by professional analysts from different fields. Due to the intertwining of business and industrial control systems and integration of different information and communication technology, it is important that the demonstration of attacks is accurate and consistent as far as possible and has a clear informative value.

In critical infrastructure, we come across diverse and often less familiar systems. At the same time, the availability as an attribute of information security is an extremely important factor; therefore, it is necessary to focus on the security perimeter. By modelling the information attacks using the presented ESM, we can successfully and effectively establish and maintain perimeter based security.

The ESM enables better coordination and management of the analysts who model the attacks, contains a larger set of data, which is adequately divided into individual sections, and also presents the course of the attack in a far less consuming manner. The model offers a number of development opportunities, especially in connection with other, publicly available databases. The model can also be used for performing security analysis of information attacks, information attack management and the integration of the model in the exploitation tools.

The majority of the interviewees are convinced that enhanced modelling using ESM will lead to improved, innovative and more detailed methods for resolving vulnerability and weaknesses difficulties and subsequently attack prevention.

The model provides more centralised information about potential attacks, thus making them more accessible to the user.

In the future, we should consider designing a software framework for attack modelling based on the presented ESM and test the model on a wider set of complex real life examples of critical infrastructures. For the purpose of creating attacks, it would be reasonable to integrate a database with known vulnerabilities and other associated attributes and manage a catalogue with offensive techniques, tactics, and procedures, which would be based on the analysis of previous threats.

REFERENCES

Alcaraz, C., and Lopez, J. (2012). Analysis of requirements for critical control systems. *International journal of critical infrastructure protection*, Vol. 5., Iss. 3-4, pp, 137-145.

Alcaraz, C., and Zeadally, S. (2015). Critical infrastructure protection: Requirements and challenges for the 21st century. *International journal of critical infrastructure protection*, Vol. 8, Pages 53–66.

Ariss O., Wu J., and Xu D. (2011). Towards an Enhanced Design Level Security: Integrating Attack Trees with Statecharts. *Fifth International Conference on Secure Software Integration and Reliability Improvement,* 1–10.

Bistarelli, S., Fioravanti, F., and Peretti, P. (2006). Defense trees for economic evaluation of security investments. *The First International Conference on Availability, Reliability and Security,* 416–423.

Bobbio, A., Egidi, L., Terruggia, R., Ciancamerla, E., and Minichino, M. (2013). Weighted attack trees for the cybersecurity analysis of Scada systems. *3rd International Defense and Homeland Security Simulation Workshop, DHSS 2013*, pp. 33-40.

Buckshaw, D.L., Parnel, G.S., Unkenholz, W.L., Parks, D.L., Wallner, J.M., and Saydjari, O.S. (2005). Mission Oriented Risk and Design Analysis of Critical Information Systems. *Military Operations Research*, V10 N2.

Chang, Y.H., Jirutitijaroen, P., and Ten, C.W. (2010). A Simulation Model of Cyber Threats for Energy Metering Devices in a Secondary Distribution Network. *5th International Conference on Critical Infrastructure*, 1–7.

Council Directive 2008/114/EC. Retrieved from: http://eur-lex.europa.eu/LexUriServ/

Edge, K. (2007). The Use of Attack and Protection Trees to Analyse Security for an Online Banking System. *Proceedings of the 40th Hawaii International Conference on System Sciences,* 144b–144b.

Falliere, N., Murchu, L., Chien, E. (2011). W32. Stuxnet Dossier. Symantec Security Response.

Fovino, I. N., Masera, M., and De Cian A. (2009). Integrating cyber attacks within fault trees. *Reliability Engineering & System Safety, 9,* 1394–1402.

Hong, J. and Kim, D.-S. (2012). HARMs: Hierarchical Attack Representation Models for Network Security Analysis. 2012.

Hurst, W., Merabti, M., Fergus, P. (2014). A survey of critical infrastructure security. *Critical Infrastructure Protection VIII, IFIP Advances in Information and Communication Technology*, Vol. 441, pp 127-138.

Ivanc, B., Klobučar, T. (2014). Attack modeling in the critical infrastructure. Elektrotehniški vestnik, Vol. 81, Iss. 5, pp. 285-292.

Ivanc, B., Klobučar, T. (2015). Use of the enhanced structural model for attack analysis and education. *Proceedings of the NATO Advanced Research Workshop on Managing Terrorism Threats to Critical Infrastructure - Challenges for South Eastern Europe*, 67-75.

Khand, P. A. (2009). System level Security modelling using Attack trees. *2nd International Conference on Computer, Control and Communication,* 1–7.

Kert, M., Lopez, J., Markatos, E., and Preneel, P. (2014). State-of-the-art of Secure ICT Landscape (Final, Version 1), NIS Platform, Working group 3 (WG3).

Kim, D.-Y. (2014). Cyber security issues imposed on nuclear power plants. *Annals of Nuclear Energy*, Vol. 65, pp. 141-143.

Kordy, B., Mauw, S., Radomirovic, S., Schweitzer, P. (2011). Foundations of Attack-defense trees. In P. Degano, S. Etalle, J.D. Guttman (Eds.), *Formal Aspects of Security and Trust* (pp. 80–95). Berlin: Springer.

Langner, R. (2011). Stuxnet: Dissecting a cyberwarfare weapon. IEEE Security and Privacy, Vol. 9, Iss. 3, pp. 49-51.

Li, W., Huang, J., and You W. (2010). Attack Modelling for Electric Power Information Networks. *International Conference on Power System Technology,* 1–5.

Lopez, J., Nielsen, J., Hemmes, J., and Humphries, J. (2012). Using Attack Trees to Assess Security Controls for Supervisory Control and data Acquisition Systems (SCADA). *Proceedings of the 7th International Conference on Information Warfare and Security*, Academic Conferences Limited, pp. 166-177.

Piètre-Cambacédès, L., and Bouissou, M. (2010). Beyond attack trees: dynamic security modelling with Boolean logic Driven Markov Processes (BDMP). *European Dependable Computing Conference*, 199–208.

McDaniel, P., and McLaughlin, S. (2012). Structured security testing in the smartgrid. *5th International Symposium on Communications Control and Signal Processing*, 1–4.

Mouratidis, H., and Giorgini, P. (2007). Security Attack Testing (SAT) – testing the security of information systems at design time. *Information Systems* (32), 1166-1183.

Roy, A., Kim, D.S., and Trivedi K. (2012). Scalable Optimal Countermeasure Selection using Implicit Enumeration on Attack Countermeasure Trees. *42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 1–12.*

Schneier, B. (1999). Attack trees: Modeling security threats. *Dr. Dobb's Journal.*

Stouffer, K., Falco, J., and Scarfone, K. (2011). *NIST Special Publication 800-82. Guide to Industrial Control Systems (ICS) Security.*

Wang, J., Phan, R.C., Whitley, J.N., and Parish, D.J. (2010). Augmented Attack Tree Modelling of Distributed Denial of Services and Tree Based Attack Detection Method. *10th International Conference on Computer and Information Technology,* 1009–1014.

Zhao, S., Li, X., Xu, G., Zhang, L., and Feng, Z. (2014). Attack Tree Based Android Malware Detection with Hybrid Analysis. *IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, 1-8.