Journal of Computer Science 2012, 8 (11), 1834-1838
ISSN 1549-3636
© 2012 Science Publications
doi:10.3844/jcssp.2012.1834.1838 Published Online 8 (11) 2012 (http://www.thescipub.com/jcs.toc)

Classification of Attacks on Embedded Systems

[1]Noura Ouerdi, [1]M'hammed Ziane, [1]Abdelmalek Azizi and [2]Mostafa Azizi

[1]Department of Mathematical and Computer, Lab. ACSA, Faculty of Sciences,
Mohammed First University, Oujda, Morocco
[2]Department of Computer, Lab. MATSI, ESTO, Mohammed First University,
ENSAO BP 669 Complexes Universities Al Qods Oujda 60000, Oujda, Morocco

Received 2012-06-13, Revised 2012-07-20; Accepted 2012-08-07

ABSTRACT

The current growth of the number of Embedded Systems (ES) and their use in sensitive systems attract the attention of attackers. They are exposed to real threats and incur significant risks especially when they are involved in specific military or industrial systems. It is important to study the subject from different angles to understand, predict and learn how to protect our products against these attacks. So, evaluators of Embedded Systems (ES) need to organize structure and classify attacks in order to choose the relevant test sets. To do this, we propose a new classification of attacks on Embedded Systems, based on Classification Tree Method (CTM). As a result, thanks to our proposal, we were able to classify the various attacks en Embedded Systems (ES) to generate test cases automatically and usually select the relevant test cases. This allows to properly evaluate Embedded Systems (ES) especially since this type of system is very critical and requires a systematic evaluation.

Keywords: Embedded System (ES), Attack, Classification Tree Method (CTM), Security

1. INTRODUCTION

An ES is a small specialized system computer, which is completely encapsulated in a device. The security of this type of system is a common topic and this can quickly become a problem, even bigger than the computer's security. One reason for the lack of security is the constraint of hardware devices in the application of security measures. For instance, a mal-packet can exploit memory-related vulnerabilities and use existing application code in a sensor without disrupting the sensor's functionality (Gu *et al*., 2011). The other reason is the cost of security. So the question is: how to effectively test and be sure that the ES behaves correctly? A trivial solution is to construct representative and meaningful classifications for all attacks. The idea is to reduce considerably the possible cases by constructing classes of attacks so that the test will take one element of each class. To improve verification techniques, we propose in the following new classification based on CTM. This classification makes the evaluation of ES more systematic and also allows the selection of attack test cases via Classification Tree Editor (CTE) tool.

The remainder of this article is organized as follows: First, we present materials and methods where we treat the old taxonomies. Then, we analyze these old classifications and we propose, as result, our new classification based on CTM method. Finally, we conclude the article.

2. MATERIALS AND METHODS

According to (Bishop, 1995), six factors are used to classify the attacks:

- Nature: The nature of the vulnerability
- Timing: When the vulnerability was introduced
- Area of operation: What is gained through the exploitation
- Effect: Which can be affected by the vulnerability
- The minimum number of components required to exploit this vulnerability
- Source of the vulnerability

Corresponding Author: Noura Ouerdi, Department of Mathematical and Computer, Lab. ACSA, Faculty of Sciences, Mohammed First University, Oujda, Morocco

In his classification, Bishop focused more on vulnerabilities rather than attacks themselves which is not the goal of taxonomy. However, it presents a good background for proposing a new taxonomy.

Howard (1997) presents a taxonomy which takes into the motivation and objectives of the attacker. This taxonomy consists of five steps:

- The attacker: Who can launch an attack
- Tools used by the attacker to access the system
- Access to the system by exploiting the design, implementation or configuration of the system
- The result
- The objectives of the attacker

Howard was interested to the process of an attack from the first step "attackers" to the last one "objectives" and not to the attack.

Lough (2001) has proposed a new taxonomy called Validation Exposure Randomness Deallocation Improper Conditions Taxonomy (VERDICT) (Lough, 2001) which is based on the characteristics of attacks:

- Improper validation: Incorrect validation results
- Improper exposure
- Improper randomness
- Improper deallocation: When the information is not properly deleted

In majority, Lough's taxonomy is general and does not include attacks in term of worms, viruses, Trojans. In addition to attacks in terms of time that are crucial for real time systems.

The classification of (Ravi *et al*., 2004) depends on the objective and the method used by attacks. Attacks are classified into four categories based on the final goal of the attack (Grochtmann *et al*., 1995): Cloning, Theft-of- Service, Spoofing and Feature unlocking. The second level of classification is the functional objective of the attack. There are attacks against privacy; attacks against integrity and attacks against availability. The third level of classification is based on the method used to execute the attack, i.e., physical, side-channel and software attacks. The **Fig. 1** describes the Ravi's classification.

Ravi considers the vision of the attacker and not that of ES. Such approaches often ignore some important features of the attacks, as seen by system administrators.

2.1. Analysis of Older Taxonomies

The older taxonomies are not really suitable for the assessment of ES security. The reasons can be broadly summarized in the following points (Ravi *et al*., 2004):

- They usually consider the vision of the attacker and his objectives
- They are not accompanied by selection and generation of test cases

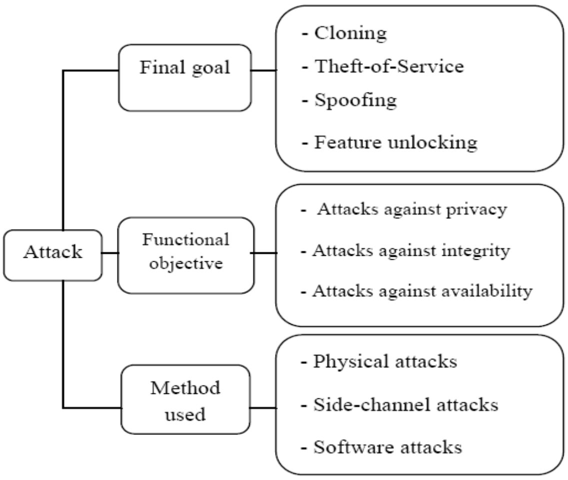

Fig. 1. Taxonomy of Ravi

For these reasons, we thought of a new classification based on the classification Tree. The main goal is to provide ES test cases that are relevant and representative of different attacks.

2.2. New Classification based on Classification Tree Method (CTM)

Classification trees have been introduced in the early 90s by Grimm and Grochtmann for presentation of test cases (Grochtmann *et al*., 1995). The construction of classification trees is implemented by the CTM method which is derived from the Category Partition Method method (CPM) (Ostrand and Balcer, 1988). The CTM provides a formal method to represent test cases graphically. It can be used to transform the specifications of a problem to a set of test cases.

The test execution is a crucial step in the process of a system developing. However, the planning of these tests often raises the same questions:

- What is the number of test cases to run
- How can we avoid applying the same tests (same class)
- How testers know that the test case is relevant

Everyone can confront these issues and it would be interesting to know that the CTM method answers to all these questions. Indeed, it offers a systematic procedure to create test cases specific to the problem.

The basic idea of the CTM method is to ignore the test data and to separate the input domain of the test into distinct subsets (called class) according to the aspects considered relevant by the tester. Then, the test cases are generated by combining classes of different classifications.

2.3. Classification Tree Editor (CTE) Tool

CTE tool is based on CTM method. It supports systematic test case determination for software testing, verification and reliability (Grochtmann and Grimm, 1993). CTE tool provides a friendly and easy interface that allows you to draw the tree based on class, classification or composition. Once the tree is created, it is possible to generate test cases automatically, just type the rule in the "Test case Generator Editor" window and Test case group will be generated in higher quality and in reduced time. Therefore the final interface will contain the tree drawn in the middle of the interface. Below the window, you will have the list of generated test cases.

Despite CTE tool is easy application, it is exceptionally helpful in analyzing and verification of the system. It is successfully used in various domains especially in safety and security area like Intrusion Detection Systems (IDS) verification process (Gadelrab *et al*., 2007).

In this study, we exploit the usability of CTE editor to analyze ES and generate appropriate test cases.

3. RESULTS

The complexity of ES consisting of hardware and software is increasing and it poses a challenge in verifying their correctness. Then, to be sure that the ES works correctly, the evaluators must test the system regularly, hence the need for automating the generation of test cases. To do this, it was first necessary to determine the dimensions of an attack on ES.

Indeed, by analyzing the previous taxonomies including the taxonomy of attacks on Intrusion Detection System (Gadelrab *et al*., 2007), we were required to adapt it to the ES. Therefore, we have defined different dimensions such as time since the ES is a real time system. We also took into account the attacks in terms of analyzing of energy consumption.

Our classification is based on five dimensions. These dimensions are:

- Means by which the attack was carried out; here, the means can be
- Consumption: This dimension is taken into account by the covert channels. In this case, the attacker makes observations of system behavior in terms of energy consumption and execution time for detecting information. Indeed, the particularity of an ES is that the analysis of its execution time or its power consumption may help the attacker to find the information. If we take as example the smart card, the card consumption depends on the values manipulated by the card (Giraud, 2007)
- Action: An action performed by an attack can be execution of a command, de-packaging, probing or use of electromagnetic fields
- Network: This mean concerns the four layers: Application, logical, transport and network layer
- Target: It may cover the hardware or software
- Hardware: An attack can target the various system components including memory, CPU, input/output control units and clock. Since an ES is a real time system. Therefore, a simple clock delay can cause a system malfunction. This is the case of systems embedded in planes, satellites
- Software: The attack can be related to the IP address, denial of service or Operating Systeml
- Source of the attack (Remote or local)
- Privilege: We distinguished four privilege classes aimed by the attacker. The "root","user","system" and "none" privilege
- Vulnerability: The exploited vulnerabilities are related to configuration, implementation or design;

In what follows, we use this classification to present a simple approach for selecting test cases. For this objective, we propose to use the CTM method.

4. DISCUSSION

We can note by CTM/ES, the classification tree specific to ES (Krupp and Muller, 2009). The classification tree has several advantages. First, the identification of all possible test cases and the selection of relevant test cases are done in a systematic manner, which makes easier its management and helps to reduce or eliminate certain errors. Moreover, its graphical representation improves visualization and facilitates communication between people who build the specification, both who are involved in developing and who manage the testing. To do this, we use the CTE tool.

Thus, from our new classification, we generate all possible combinations of subclasses using the CTE tool. These combinations represent test cases for possible attacks. More precisely, the CTE offers a simple and powerful formalism for constraints expression by combining some rules which include some sub-ones. Between brackets (under a predicting form), some connectors such as:

- and (*)
- or (+)
- no (NOT)

We propose a rule which represents Physical Attacks:

$$\text{Local} * (\text{CPU} + \text{Clock}) * \text{Privilege\_None} * (\text{De} - \text{packaging} + \text{Probing}) * (\text{Design} + \text{Implementation})$$

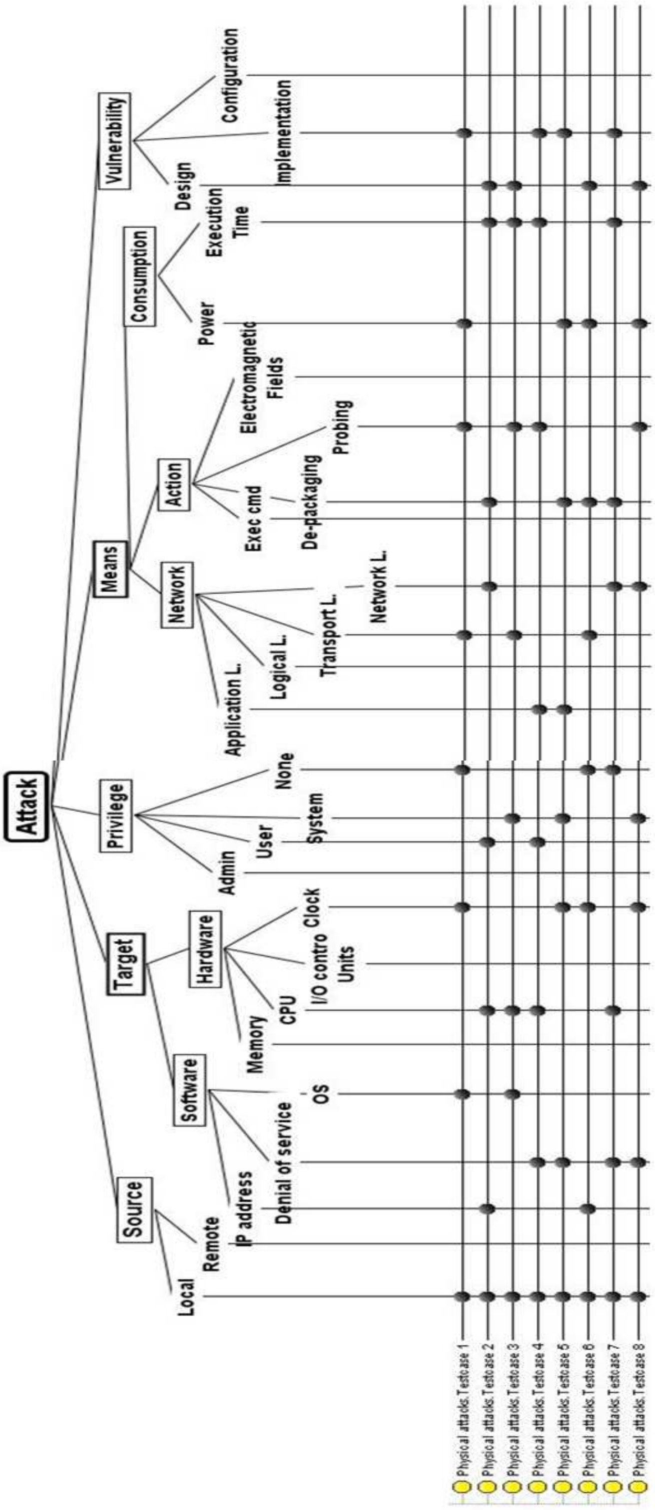

Fig. 2. Classification tree of ES attacks and generated test cases via CTE

With this rule, we have eight test cases, generated automatically (**Fig. 2**).

We can propose rules for other attacks like covert channels:

$$\begin{gathered}(\text{Local} + \text{Remote}) * (\text{I/O control Units} + \text{Clock}) \\ *(\text{Admin} + \text{System}) * (\text{Transport L.} + \text{Network L.}) \\ *(\text{Probing} + \text{Electromagnetic\_fields}) * \\ (\text{Power} + \text{Executiontime}) * \\ (\text{Design} + \text{implementation} + \text{configuration})\end{gathered}$$

With this complex rule, we have generated one hundred ninety-two test cases.

Now, we are able to generate test cases automatically in order to test our ES against potential attacks. Simply, we should determine the generation rule from the tree and the rest will be made by the CTE tool.

5. CONCLUSION

This study presents two approaches for improving the ES evaluation process:

- A systematic method of generating test cases
- Selecting test cases based on an appropriate attacks classification

We can, thus, have a better knowledge of space attacks and improve our understanding of attack instances, including the most recent and therefore have a security level conditioned by the verification phase.

Regarding the design phase, usually, the security of ES is not taken into account in an earlier stage and it is difficult to implement it once the product is achieved. Security must be integrated into the product during the design phase (Design for Security). However, the implementation of security measures is not enough. It must also verify the effectiveness of these measures and check hidden threats. The product must be regularly monitored and updated to have the greatest impact against attacks. In addition, the security measures should be taken upon detection of a problem (Grand, 2004). In the same context, (Weingart, 2000) describes the various mechanisms ranging from the easiest and cheapest to the most difficult and extremely expensive. To conclude, the definition of new methodologies for safety assessment, which takes into account the changing of security nature, could ensure security of our products.

6. REFERENCES

Bishop, M., 1995. A taxonomy of UNIX system and network vulnerabilities. University of California at Davis, Davis, CA.

Gadelrab, M.S., A.A. El Kalam and Y. Deswarte, 2007. Defining categories to select representative attack test-cases. Proceedings of the 2007 ACM Workshop on 5 Quality of Protection, (QoP' 07), ACM Press, USA., pp: 40-42. DOI: 10.1145/1314257.1314270

Giraud, C., 2007. Attacks against cryptosystems and board-related measures. Ph.D Thesis, Superior Normal School, University of Versailles Saint-Quentin-en-Yvelines, Paris, France.

Grand, J., 2004. Practical secure hardware design for embedded systems. Proceedings of the 2004 Embedded Systems Conference, Mar. 29-Apr. 1, San Francisco, California.

Grochtmann, M. and K. Grimm, 1993. Classification trees for partition testing. Software Test. Verific. Reliab., 3: 63-82. DOI: 10.1002/stvr.4370030203

Grochtmann, M., J. Wegener and K. Grimm, 1995. Test case design using classification trees and the classification-tree editor CTE. Proceedings of the 8th International Software Quality Week, (ISQE' 95), pp: 1-11.

Gu, Q., C. Ferguson and R. Noorani, 2011. A study of self-propagating mal-packets in sensor networks: Attacks and defenses. Comput. Security, 30: 13-27. DOI: 10.1016/j.cose.2010.10.002

Howard, J.D., 1997. An analysis of security incidents on the internet 1989e1995. Ph.D Thesis, Carnegie Mellon University.

Krupp, A. and W. Muller, 2009. Systematic model-in-the-loop test of embedded control systems. IFIP Adv. Inform. Commun. Technol.

Lough, D.L., 2001. A taxonomy of computer attacks with applications to wireless networks. Ph.D Thesis, Virginia Polytechnic Institute and State University.

Ostrand, T.J. and M.J. Balcer, 1988. The category-partition method for specifying and generating fuctional tests. Commun. ACM, 31: 676-686. DOI: 10.1145/62959.62964

Ravi, S., A. Raghunathan and S. Chakradhar, 2004. Tamper resistance mechanisms for secure embedded systems. Proceedings of the 17th Internatioanal Conference of VLSI Design, (VLSID' 04), IEEE Xplore Press, pp: 605-611. DOI: 10.1109/ICVD.2004.1260985

Weingart, S.H., 2000. Physical security devices for computer subsystems: A survey of attacks and defences. Proceedings of the 2nd International Workshop on Cryptografic Hardware and Embedded Systems, (CHES' 00), ACM Press, London, pp: 302-317.