AIDD: A novel generic attack modeling approach

Samih Souissi, Ahmed Serhrouchni

AIDD: A novel generic attack modeling approach

Samih Souissi, Ahmed Serhrouchni
INFRES Department
Telecom ParisTech
Paris, France
{souissi, serhrouchni}@telecom-paristech.fr

***Abstract*— In recent years, information systems have become more diverse and complex making them a privileged target of network and computer attacks. These attacks have increased tremendously and turned out to be more sophisticated and evolving in an unpredictable manner. This work presents an attack model called AIDD (Attacks Identification Description and Defense). It offers a generic attack modeling to classify, help identify and defend against computer and network attacks. Our approach takes into account several attack properties in order to simplify attack handling and aggregate defense mechanisms. The originality in our work is that it introduces a target centric classification which increases the level of abstraction in order to offer a generic model to describe complex attacks.**

***Keywords*— attack modeling, attack taxonomy, attack classification, attack detection, network and web attacks, defense mechanisms**

I. INTRODUCTION

Over the past few years, information technology has become widespread and heterogeneous. Along with this rapid development, attacks on information systems have increased greatly and have become not only numerous and diverse but also complex and sophisticated. This situation has raised several security challenges. With the growing complexity of attacks and the advent of new ones, many solutions such as intrusion detection and prevention systems (IDS/IPS) [1] and web application firewalls (WAF) [2] have been proposed in order to counter these attacks.

However, these systems are not always able to detect attacks and can lead to false positives or false negatives. Indeed, these solutions tend to be based on static rules and able to detect only specific attacks or anomalous behaviors that are already known. Therefore, in order to ameliorate detection and to know how to respond to complex attacks, solutions need to know how to identify attacks and how to assign appropriate defense mechanisms. As it is hard, on the one hand, to list all existing attacks, and on the other hand, to detect new and complex attacks, an obvious solution would be to create a relevant classification which represents all current attacks. Although, several classifications of vulnerabilities (CVE [3], OSVBD [4]) and of attacks (OWASP [5], WASC [6], CAPEC [7]) exist and are supported by many security tools, no attack classification is widely used or considered as a standard. In addition, existing attack taxonomies are not generic and are not commonly accepted or referenced. Existing classifications are not evaluative and can no longer be interesting when new and complex attacks or systems appear. In our context, attack modeling is crucial to the detection process. It is also closely related to the choice of implemented rules and attack detection parameters within IDS or WAF. The idea behind our approach is to build attack classes and to define unique describing parameters for every attack. This brings a level of abstraction that will make detection of complex and new attacks more feasible and simplify rules defining process.

The objective of this paper is to present not only an attack taxonomy but also an evaluative model for attack description that allows having a common background and language to describe these attacks and setup the appropriate defense mechanisms. Besides, our proposition offers a better understanding of attacks and a decision-making tool to detect attack, enhance security and increase systems' robustness.

The remainder of this paper is organized as follows. Section II details the related work concerning existing attack classifications. We present, in section III, our proposition describing the methodology followed and the attack model. In section IV, we expose a use case of AIDD classification showing how it can bring a higher level of abstraction and better describe attacks. Finally, Section V presents the conclusion and perspectives for future work.

II. RELATED WORK

In order to ensure network and computer security, it is fundamental to identify attacks and vulnerabilities. Given the inability to exhaustively examine all existing attacks, researchers have done much work in the field on classifying them. Research initially focused on vulnerabilities instead of attacks [8]. Then, early attack classifications aimed at one attack dimension [9] [10]. Later, studies have been oriented toward multidimensional taxonomies that are more suitable to describe attacks.

Hansman & Hunt [11] were the first to introduce the concept of dimensions with several levels and a description of each one. They classify attacks into four main dimensions. The first one is "Attack Vector" or the principal means by which the attack reaches the target. The second dimension is "Attack Target" that can be hardware, software, network, protocol, etc. The third dimension consists of the "Vulnerability" exploited during the attack. The fourth and final dimension concerns "Attack effects" which are the results or the impacts of the attack itself. These dimensions are subdivided to provide more specificity. Overall, they give a good overview of attacks and methods available. This taxonomy is the first to introduce the

concept of dimension to classify attacks. This approach helps identify attacks and better describe them. It helps improve computer and network security and add coherence while describing attacks. However, it is not complete and extra dimensions could be added to improve the taxonomy, such as defensive ones. Besides, Hansman and Hunt state the need for future work to improve blended attacks classification.

In [12], Gadelrab et al. propose a classification for IDS evaluation. Their attack classification is based on five dimensions: "Source" which indicates the location where attack is launched from, "Privilege" which indicate the access gained during the attack, "Vulnerability" which indicates the flaw related to the attack from an evaluation perspective for referencing purposes, "Means" via which the attack is initiated and "Target" of the attack. This classification includes observable characteristics of the attack from the evaluator point of view. It allows a good description of the attacks from different angles. However, it may present problems of mutual exclusion. Moreover, it does only consider privilege escalation and probe as an attack result and it does not consider the defense mechanisms.

Simmons et al. [13] propose a cyber-attack taxonomy called AVOIDIT. Five categories characterizing the nature of an attack are used: "Attack Vector" that is the path by which an attacker can gain access to the host, "Operational Impact" containing a list of results of the attack to provide high level information, "Defense Mechanism" which contains strategies used by defender, "Informational Impact" which classifies the effect of the attack on information, and "Attack Target". AVOIDIT is an interesting taxonomy that takes into account defense mechanisms able to classify blended attacks. It also brings the appropriate information to help the defender make an educated decision when defending against attacks. The limitations of taxonomies are the lack of defense strategies and the fact that the defense aspect is just used for informative purposes not during the time attack impacts. It also doesn't consider attacks with no result.

In [14], Wu et al. present a response-oriented taxonomy of attacks. It is based on three dimensions: "Source" which is the origin of the attack, "Techniques" which are the methods adopted by attackers and "Results" of the attack. Based on this taxonomy, Wu et al. build corresponding relationships between attack and response. This taxonomy is one of the most interesting since it is directed towards the response to the attacks after discovery. In fact, it can describe the attacks and is quite flexible. This is a good basis for response-oriented attack taxonomy. However, this taxonomy does not take into account the target type for its decision. We find also that blended attacks are difficult to classify and the technical dimension of the taxonomy should be refined.

Since multidimensional classifications are able to describe precisely attacks from different angles, they have been usually used in practice and thus they are interesting to define an attack model. However, most of the taxonomies studied are elaborated from the viewpoint of attackers and are not necessarily suitable to assign the appropriate defense mechanisms as more information from a target (or a defender) perspective is needed. In fact, a taxonomy depends on the application and purpose which it was created for. Sometimes, the definition of classes and subclasses is unclear creating problems of mutual exclusivity. Moreover, complex attacks are not easily classified within the previously exposed taxonomies.

III. PROPOSAL

Our model concept is based on what have already been proposed as attack classifications. In our study, we are interested in identifying recurrent classes in existing taxonomies and adapt them to our context. We have studied 23 different attacks classifications. While considering all different approaches, we have identified relevant dimensions. These dimensions help make broader classifications focusing more on a security context than the simple attack by itself.

Based on this study and in order to model attacks, we define a response-oriented classification. As shown in Fig.1, our model takes events as inputs and detects attacks parameters. It provides, as output, an attack classification and defense mechanisms. The choice of appropriate mechanisms to prevent this category of attacks is done by a matching module.

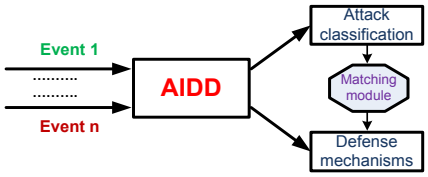

Fig. 1. AIDD Model

As we have shown previously, the existing taxonomies are not able to meet the requirements of the model that we intend to propose. These requirements are: aggregation of attacks (classification must find common characteristics of attacks t group them), completeness (classification must be complete, containing all currently known attacks), stability (new attacks must not challenge the classification), flexibility and scalability (classification must be flexible enough to adapt to changes of topology, architecture, new attacks...etc.), optimal recovery of interclass attacks (given the impossibility of having a classification respecting mutual exclusivity, our classification must cover attacks that can belong to several classes at once) and unified defense mechanisms (classification must involve unified response for each class of attacks).

In the following, we define the different classes of attack classification and a set of various defense mechanisms that can be assigned.

A. Attack classification

The aim of this classification is to provide a generic model for attacks' description to help detect and provide the appropriate response mechanisms. Based on the preliminary study, we define a classification that allows satisfying the requirements and specifications mentioned above. Fig.2 shows the two first levels of attack classification. It is composed of four classes: Source, Target, Vector and Result.

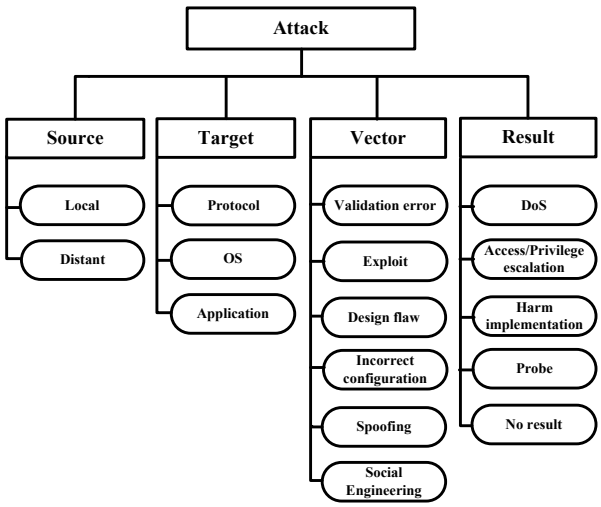

Fig. 2. The two first levels of attack classification

1) Source Class

The source of the attack is the specified location from where the attack is initiated. There are two sub-classes: "local", when attack is initiated from the target itself and "distant", when attack is initiated outside of the target. This last subclass can be subdivided to local network attacks or distant network.

2) Target Class

It indicates system's component that is targeted by the attacker. Attacks are various and can aim for different types of hosts. The attack can target a particular "protocol" through certain vulnerabilities. The attack can also target specific software; the "application" can be a client application that is specific to one user or a server application that hosts multiple users. Finally, the attack can also target a particular vulnerability within an operating system.

3) Vector Class

Vectors are methods used by the attacker as they occur at the victim side. This class contains also vulnerabilities because exploiting vulnerabilities is required to launch an attack. It is composed of: Validation error, Exploit, design flaw, spoofing, incorrect configuration and social engineering.

"Validation error" happens when a system fails to validate user input to a certain program. It is an error due to wrong requests that are received by the system and are not defined or verified. It can be a buffer overflow attack or boundary conditions happening when a process tries to read or write beyond authorized limits, or when resources are exhausted. It can also be malformed input which is due to a process that accepts an invalid entry or syntactically invalid input field. The other sub-class is "exploit" that consists of vulnerabilities or undefined state which causes performance degradation or system compromise. It can be an exception which is caused by the failure of managing an exception that is generated by a function or a module, a race condition which is an error that occurs during the time window between two operations, a serialization error due to bad serialization operations or an atomicity error when partially modified data structure is being used by another process, or a finished process with a partial data modification instead of atomic modification. Another vector is "design flaw" where attacks target a design or a protocol structure. It is related to exploiting an erroneous conception of a solution or a network protocol and includes for example flooding attacks. "Incorrect configuration" contains vulnerabilities from a faulty software configuration that can lead to attacks. "Spoofing" attacks can be another vector where a malicious user impersonates a legal one to hide its identity or gain access. The last vector is "social engineering". It consists of attacks that exploit the human aspect of information systems by manipulating people into performing wrong actions or revealing information.

4) Result Class

This class contains the final result of the attack. It helps gather information to better describe attacks from an impact perspective. We provide a list of mutually exclusive results. This class is composed of: denial of service, access/privilege escalation, probe, and harm implementation. We also consider the case when the attack didn't succeed.

"Denial of service" (DoS) is an attack that causes a denial of access to a resource or a service for a victim. It can be host based (attacking a specific computer system), network based (targeting a complete network to prevent the network from working normally) or distributed (using multiple vectors to reach the attacker's purpose). "Access/Privilege escalation" happens when the attacker obtains access to services at the system level of the victim or the attack causes a total system control. It involves getting rights using illegitimate manners. The attacker can get access anonymously and elevate its privileges to have user or administrator rights. "Harm implementation" can consist of installing a malware that can be the launching platform of an attack (virus, Trojan, worm, spyware), executing a code remotely to corrupt the target, a resource misuse or theft which is an unauthorized use of resources extended and using privilege gained for abusive action, or an Information corruption when information is corrupted or modified. "Probe" consists of a scan or any other activity that leads to a disclosure of information or system properties. This information can be related to user, network, system, hosts and setting. To be complete, we added "No result" sub-class in case of attack failure.

B. Defense mechanisms

We expand the classification exposed to add a defense class. We outline the different defense mechanisms that can be deployed before, during and after the attack. They are composed of detection, prevention, response, tolerance and awareness. A combination of defense mechanisms can be used when trying to counter attacks.

Attack "Detection", can be signature-based or anomaly based and the source of the attack can be tracked. "Prevention" mechanisms avoid the occurrence of the attack by implementing anti-spoofing systems and ameliorating equipment's security (system hardening, audit). "Response" subclass contains mechanisms to respond to the cause of the attack, trying to mitigate or remediate. On the one hand, mitigation aims to reduce the attack severity. It can be:

quarantine when infected hosts are removed from network, filtering when listing the possible permitted connections and referencing, or reporting by providing report to mitigate an attack or to references of the eventual vulnerability that causes the attack. On the other hand, remediation helps correct the problematic situation by taking the appropriate steps. It can take the form of applying a Patch provided by software vendor to remove a certain vulnerability present within this software, code correction when application source code modification are performed to cease the potential exploit of a vulnerability by an attacker, or authentication to secure the access to the network, the system or the application. "Tolerance" sub-class means that system administrator accepts the vulnerability threat. The operational impact of responding to the attack is considered as not worthwhile. The last subclass that is more organizational is "Awareness". This mechanism concerns the human factor in securing information systems. The users should know how to use applications and systems in a secure manner and should be aware of social engineering attacks.

This model offers the possibility to characterize attacks with few parameters to help creating a relationship with defense mechanisms using a matching module. Thanks to this specific attack modeling, a list of defense mechanisms can be assigned. Thus, attack's response can be automated.

IV. USE CASE: MODEL APPLICATION

In this section, we illustrate how our model helps classify attacks and offer appropriate responses. We highlight how AIDD can bring a higher level of abstraction in order to better classify attacks. Our model is able to classify complex blended attacks by subdividing each step, considering each step as an attack on its own and providing for each step defense mechanisms. Thus, it helps detection and response engine to stop the attack before the occurrence of the final impact.

TABLE I. COMPLEX ATTACK

Attack	**Attack classes**			
Phase 1	***Source***	***Target***	***Vector***	***Result***
	Distant network	OS-User	Design flaw	Probe
Defense mechanism	Detection - signature based Response- reporting - Filtering			
Phase 2	***Source***	***Target***	***Vector***	***Result***
	Distant network	OS-User	Misconfiguration	Malware installation - Worm
Defense mechanism	Detection - signature based Response - Mitigation/Patch			
Phase 3	***Source***	***Target***	***Vector***	***Result***
	Local	OS-User	BoF	Resource misuse
Defense mechanism	Response - Patch/Quarantine			

In this attack scenario, the attacker compromises a host to attack a target. From this host, he performs a scan to gather information about the targets. Then he exploits a misconfiguration to install a worm. This worm will help the attacker launch a buffer overflow attack leading to a resource misuse. As shown in Table I, a complex attack is decomposed into 3 different phases. Defense mechanisms can be assigned to each phase. The model anticipates the occurrence of phase 3 if faced with the 2 first ones. Thus, AIDD model helps improve attack detection and response in such cases.

V. CONCLUSION AND FUTURE WORK

Until now, few attack classifications have taken into account the defense aspect. In this paper, we have proposed a novel attack model that ensures classifying attacks and assigning appropriate defense mechanisms. Our model will be used by network and system administrators to provide information about previous attacks. It is not only a classification but also a framework that learns from previous events and helps decide which defense mechanisms to use.

We show that our model is able to describe complex attacks and provide fitting defense. This is a good start toward a better a response oriented attack description. Our model is defined in such high level manner that he can remain stable and handle new types of attacks. It can be adapted to new topologies and can include new attack techniques. We are aware that our model can be ameliorated by including encrypted information handling and defining metrics to enhance the attack-defense matching process. The next step is to specify the matching module and define an architecture for this model.

REFERENCES

[1] Curtis A. Carver Jr., "Intrusion Response Systems: A Survey", Texas A&M University, College Station, TX 77843-3112, USA

[2] Jaeson Yoo, "A Call for Drastic Action, A Survey of Web Application Firewalls", Penta Security Systems Inc., OWASP

[3] CVE, http://www.cve.mitre.org/

[4] OSVBD, http://www.osvdb.org/

[5] OWASP, http://www.owasp.org/

[6] WASC, version 2.0, http://www.webappsec.org/

[7] CAPEC, http://capec.mitre.org/

[8] Bishop M., "A taxonomy of Unix and network security vulnerabilities", Department of Computer Science, University of California at Davis, May 1995

[9] Richard Lippmann, Joshua W. Haines, David J. Fried, Jonathan Korba, and Kumar Das, "Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation", MIT Lincoln Laboratory 244 Wood Street, Lexington, MA02173-9108, USA

[10] Ulf Lindqvist and Erland Jonsson, "How to systematically classify computer security intrusions", University of Technology Göteborg, Sweden, 1996

[11] Simon Hansman, Ray Hunt, "A taxonomy of network and computer attacks", University of Canterbury, New Zealand

[12] Mohammed EL-Sayed Gadelrab, "Evaluation des Systèmes de Détection », PhD Thesis, Toulouse University, France

[13] Chris Simmons, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Qishi Wu, "AVOIDIT: A Cyber Attack Taxonomy", Department of Computer Science, University of Memphis

[14] Zheng Wu, Yang Ou, Yujun Liu, "A Taxonomy of Network and Computer Attacks Based on Responses", 2011