A Taxonomy of Web Attacks

Gonzalo Álvarez and Slobodan Petrović

Instituto de Física Aplicada, C.S.I.C., Serrano 144, 28006 Madrid, Spain,
{gonzalo,slobodan}@iec.csic.es,
http://www.iec.csic.es/

Abstract. A new taxonomy of web attacks is proposed in this paper, with the objective of obtaining a useful security reference framework. Possible applications are described, which might benefit from this taxonomy, such as intrusion detection systems and application firewalls.

1 Introduction

By *web attacks*, we understand attacks exclusively using the HTTP/HTTPS protocol. When the web attacks recorded through the years are analyzed, it is observed that most of them are recurrent attacks. In fact, they correspond to a limited number of types of attacks. Thus, it is generally agreed that classification can help designers, programmers, and security analysts to better understand attacks and build more secure applications.

In an effort to create a common reference language for security analysts, a number of taxonomies of computer attacks and vulnerabilities have appeared in recent years [1,2,3,4,5]. The shortcoming of such taxonomies is that they often include categories unsuitable for the classification of web attacks. Even when their categories can be used in a web context, they fail to cover all the subtleties of web attacks. For example, entry point, target, HTTP Verb, and HTTP Header are web-specific categories that we consider important for a more accurate classification of web attacks, and these are not covered by general taxonomies. In addition, some categories that can also be met in general taxonomies, such as vulnerability, need to take web-specific values (e.g. Code injection, HTML manipulation).

In this paper we propose a taxonomy of web attacks, taking into account a number of important features of each attack category. The role of these features as well as their importance for each of the attack categories is discussed. The ideas about the applications that might benefit from this taxonomy are also presented.

2 Web Attack Properties

A *taxonomy* is a classification scheme that partitions a body of knowledge and defines the relationship of the objects. *Classification* is the process of using a tax-

J.M. Cueva Lovelle et al. (Eds.): ICWE 2003, LNCS 2722, pp. 295–298, 2003.

onomy for separating and ordering [2]. Satisfactory taxonomies have classification categories with the following characteristics: mutually exclusive, exhaustive, unambiguous, repeatable, accepted, and useful.

First, we introduce a novel model of web attacks based on the concept of *attack life cycle*. By attack life cycle we understand a succession of steps followed by an attacker to carry out some malicious activity on the web server, as depicted in Figure 1. The attacker gets through an entry point, searching for a vulnerability in the web server or web application, which might be exploited to defeat some security service. The vulnerability is realized by an action, using some HTTP verb and headers of certain length, directed against a given target and with a given scope. The attacker might obtain some privileges that depend on the type of attack. Our taxonomy of web attacks is based on the attack life cycle defined in this way.

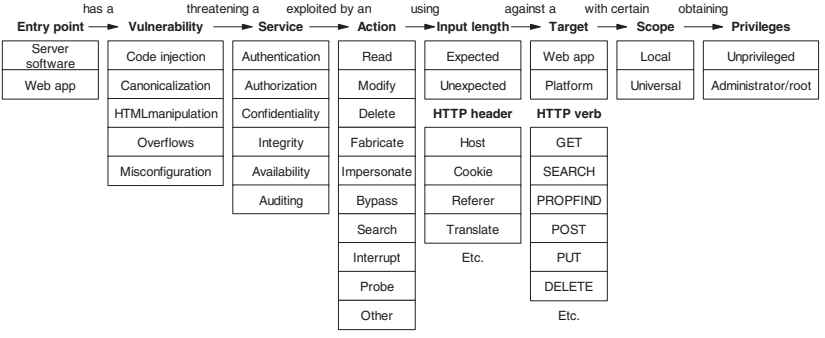

Fig. 1. Taxonomy of web attacks

At every stage of the life cycle we define the following classification criteria or classifiers:

1. Entry point: where the attack gets through. We distinguish between *web server software* attacks and *web application* attacks. The former are due to errors in the server software and, thus, they are shared by all web applications running on top. The origin of the latter depends on the web application. Those may be errors in HTML forms, client-side scripts, server-side scripts (.asp, .jsp, .php, .pl, etc.), business logic objects (COM, COM+, CORBA, etc.), SQL sentences processing, etc.
2. Vulnerability: a weakness in a system allowing unauthorized action. We define the following vulnerabilities in web applications: code injection (script, SQL, or XPath injection), canonicalization, HTML manipulation (often accomplished through URL query strings, form fields, or cookies), overflows, and misconfiguration.

3. Service (under threat): security service threatened by the attack. We distinguish between attacks against the following services: authentication, authorization, confidentiality, integrity, availability, and auditing.
4. Action: actual attack realizing the threat against the web server exploiting the vulnerability. We distinguish among actions aimed at three different objectives: server data, user authentication, and web server. Actions directed against data include *read*, *modify*, *delete*, and *fabricate*. Actions directed against authentication include *impersonate*, *bypass*, and *search*. Actions directed against the web server include *interrupt*, *probe*, and *other*.
5. Length: the length of the arguments passed to the HTTP request. We distinguish between *common-length* and *unusually long* attacks.
6. HTTP Verb: Verbs needed to perform the attack: GET, POST, HEAD, etc.
7. HTTP Header: Headers needed to perform the attack: Host, Cookie, Referer, etc.
8. Target: the aim of the attack. We distinguish between *application* attacks and *platform* attacks. In application attacks, only the application data and functionality is affected, but not the operating system resources. These attacks are typically aimed at web pages, web users, and web data. In platform attacks, the attacker usually seeks after arbitrary command execution, manipulation of machine accounts, tampering with the host's services, obtaining network information, etc.
9. Scope: impact of the attack on the web server. We distinguish between *local* (only one user or a small group of users is affected) and *universal* attacks (all users are affected).
10. Privileges: privileges obtained by the attacker after the successful completion of the attack. We distinguish between *unprivileged* and *administrative* attacks.

3 Possible Applications

This taxonomy is useful in a number of applications, especially in intrusion detection systems and in application-level firewalls.

3.1 Intrusion Detection Systems (IDS)

An intrusion detection system (IDS) detects and reports attempts to misuse or break into networked computer systems in real time [6]. One of the biggest problems faced by these systems is the huge amount of alerts that might be generated in a heavily-attacked environment in a matter of hours. It is impossible for a human operator to analyze so many reports and decide on the severity of the detected attacks to determine the action to take. This taxonomy can be used in the following way: first, the attacks are encoded into vectors using an accepted encoding scheme. Next, these vectors are processed using pattern recognition or information extraction techniques (clustering algorithms, supervised learning, etc.) in order to pinpoint the most dangerous attacks, and analyze attack trends throughout time.

3.2 Application Level Firewalls

Another approach to detect and prevent web attacks consists of using application level firewalls or, more specifically, web application firewalls. Application level firewalls are capable of processing data at the application level as well as decrypting SSL connections. An application-layer solution works within the application that it is protecting, inspecting requests as they come in from the network level. If at any point a possible attack is detected, it can take over and prevent unauthorized access and/or damage to the web server.

Classifying the attacks once they have been blocked by the firewall and deciding on their severity is crucial for a prompt and effective response. This taxonomy helps in this task, providing an exhaustive group of mutually exclusive categories under which the attacks can be unambiguously classified. Afterwards, a decision system can determine the priority of various attacks.

4 Conclusion

In this paper, a taxonomy of web attacks is proposed that intends to represent a forward step towards a more precise reference framework. An attack life cycle is defined as its base, to make it structured and logical. The properties of the most common web attacks are enumerated. Possible applications are described, which might benefit from this taxonomy, such as intrusion detection systems and application firewalls. We are working on the automated encoding of web attacks using different-length vectors and determining their priority using edit distance clustering.

Acknowledgements. This research was supported by Ministerio de Ciencia y Tecnología, Proyecto TIC2001-0586.

References

1. Cohen, F.B.: Information system attacks: A preliminary classification scheme. Computers and Security **16** (1997) 29–46
2. Howard, J.D., Longstaff, T.A.: A common language for computer security incidents. Technical Report SAND98-8667. Sandia National Laboratories (1998)
3. Lindqvist, U., Jonsson, E.: How to systematically classify computer security intrusions. Proceedings of the 1997 IEEE Symposium on Security & Privacy (1997) 154–163
4. Lough, D.L.: A Taxonomy of Computer Attacks with Applications to Wireless Networks. PhD thesis. Virginia Polytechnic Institute and State University (2001)
5. Richardson, T.W.: The Development of a Database Taxonomy of Vulnerabilities to Support the Study of Denial of Service Attacks. PhD thesis. Iowa State University (2001)
6. Northcutt, S.: Network Intrusion Detection, Third Edition. New Riders Publishing (2002)