Information Sciences 254 (2014) 19–38

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Semantic security against web application attacks

Abdul Razzaq *, Khalid Latif, H. Farooq Ahmad, Ali Hur, Zahid Anwar, Peter Charles Bloodsworth

School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad, Pakistan

ARTICLE INFO

Article history:
Received 17 May 2011
Received in revised form 4 July 2013
Accepted 6 August 2013
Available online 19 August 2013

Keywords:
Application security
Semantic security
Semantic rule engine

ABSTRACT

In this paper, we propose a method of detecting and classifying web application attacks. In contrast to current signature-based security methods, our solution is an ontology based technique. It specifies web application attacks by using semantic rules, the context of consequence and the specifications of application protocols. The system is capable of detecting sophisticated attacks effectively and efficiently by analyzing the specified portion of a user request where attacks are possible. Semantic rules help to capture the context of the application, possible attacks and the protocol that was used. These rules also allow inference to run over the ontological models in order to detect, the often complex polymorphic variations of web application attacks. The ontological model was developed using Description Logic that was based on the Web Ontology Language (OWL). The inference rules are Horn Logic statements and are implemented using the Apache JENA framework. The system is therefore platform and technology independent.

Prior to the evaluation of the system the knowledge model was validated by using OntoClean to remove inconsistency, incompleteness and redundancy in the specification of ontological concepts. The experimental results show that the detection capability and performance of our system is significantly better than existing state of the art solutions. The system successfully detects web application attacks whilst generating few false positives. The examples that are presented demonstrate that a semantic approach can be used to effectively detect zero day and more sophisticated attacks in a real-world environment.

1. Introduction

Web application security protects the confidentiality, integrity and availability of resources in cyber space. Web services and applications have shaped a new landscape of information sharing which has increased the productivity of e-business. On the other hand however, the number of cyber-threats has also increased tremendously due to the growing popularity of web applications [40,79,28,10,37]. Enormous efforts have been made to mitigate these attacks through various security mechanisms in the form of scanners, intrusion detection systems, encryption devices, and web application firewalls. These traditional security solutions often apply a signature based approach and are only effective therefore against "well-known" security flaws where signatures are already present in the solution database. Signature-based systems maintain a database of the easily identifiable "signatures" of known attacks and apply pattern matching algorithms for attack detection. Unchecked input validation is a major source of attacks at the web application level.

* Corresponding author. Tel.: +92 51 9085 2400.
E-mail addresses: abdul.razzaq@seecs.edu.pk (A. Razzaq), khalid.latif@seecs.edu.pk (K. Latif), farooq.ahmad@seecs.edu.pk (H.F. Ahmad), ali.hur@seecs.edu.pk (A. Hur), zahid.anwar@seecs.edu.pk (Z. Anwar), peter.bloodsworth@seecs.edu.pk (P.C. Bloodsworth).

0020-0255/$ - see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.ins.2013.08.007

According to the Open Web Application Security Project (OWASP) [50], four out of the current top ten vulnerabilities are related to input validation. Failure to protect web applications from invalid inputs, can often cause costly security breaches to organizations. Hacking incidents can result in the theft of sensitive data, defacement of web sites, privilege escalation, and unauthorized access to a system. A hacker might also be able to inject malicious code to bypass or modify the intended functionality of a program without the user knowing. For example, in an XSS attack, the user is deceived into clicking on a link which points to a trusted site, but it actually contains a malicious script which was written by a hacker. For example:

```
http://www.citibank.com <script>http://www.evil.com/payload?c=‘+document.cookie</script>
```

Encoding schemes also play an important role in deceiving an attack vector and commonly facilitate the malicious intention of the hacker. For example in case of a directory traversal attack the string ‘../’ is considered as a malicious string but it can also be presented in hexadecimal encoding “%252E%252E%252F”. Similarly there are hundreds of ways to launch SQL Injection attack. A few samples are shown in Fig. 1.

Traditional security solutions such as web scanners, provide the first line of defense against attack and detect well-known security flaws using threat signatures. Scanners lack semantics and are thus unable to make intelligent decision upon data leakage or business logic flaws [24]. This frequently results in false alarms being raised and such approaches also fail to detect novel and critical vulnerabilities [45]. Signature based solutions maintain a white and black list of processes. These contain the signatures of benign inputs and also those of malicious attack vectors. The use of such lists requires continuous updating of threat signatures. Failing to do so may result in limited or no protection against zero day attacks and the generation of too many false positive and false negative alerts [56]. Furthermore, most network solutions ignore the payload and only scan the headers of a user request. Due to a lack of information regarding the application context, network solutions are commonly ineffective against application level attacks. A huge amount of effort has been expended by the security industry to mitigate these attacks through various state of the art security mechanisms in the form of scanners, intrusion detection systems, encryption devices, and firewalls. These measures have unfortunately so far been unable to achieve the security level that is necessary for web applications. There is a clear need therefore for a different approach.

Semantic-based systems are designed to intelligently understand the application’s context, the data and the nature of possible attacks. Such systems validate the input to it at both the syntactic and semantic levels. Syntax based validation commonly applies size or content restrictions. Semantic based validation on the other hand mainly focuses on specific data types, formats and an understanding of potentially malicious commands with respect to their context and likely consequences. In recent years such semantic approaches have shown promise in terms of providing rich representations of web application domain knowledge [29]. Viswanathan and Krishnamurthi [77] proposed a personalization approach for making semantic relationship paths by capturing the user’s interest level in various domains through their web browsing history. Rubiolo et al. [62] presented a technique that was based on an Artificial Neural Network model, for knowledge discovery through ontology matching on the Semantic Web. The concept of semantic annotations was applied recently to develop an identity management system which was deployed as a firewall for protecting digital assets [1].

Semantic annotations and ontologies can also help us to capture the precise specification of a security model in order to improve its prevention and reaction capabilities [33,38,75,43,55]. For example, intrusive behaviors can be systematically modeled to the required level of granularity. However the actual power and utility of this is determined by the expressiveness of the ontology in modeling attack scenarios in a generalized way and at a certain level of abstraction. The ontology models developed in the prior art mainly focused on lightweight representations of the attributes of the attacks in a taxonomic structure. These models seriously lack the necessary ontological modeling and subsequent reasoning capabilities. Moreover such systems focus on applications within the network layer or access control mechanisms for digital assets.

The method proposed in this paper for detecting and classifying web application attacks differs from current signature-based security solutions. We have developed an ontology based technique that specifies web application attacks using the

```
" ‘OR 1=1’
" OR    1      =%091
" OR ‘Password’ = ‘Password’
" OR 1%3D1
" OR 1 /* comments */ = 1
" OR  2 > 1
" OR ‘Script’ > ‘R’
" OR ‘Script’ < ‘Y’
" OR ‘Script’ = N’ Script ’
" OR ‘Script’ = ‘Scri’+’pt’
" OR ‘Script’ LIKE ‘Scr%’
" OR ‘Script’ IN (‘Script’)
" OR ‘Script’ BETWEEN ‘R’ AND ‘T’
```

Fig. 1. Heterogeneity in a tautology attack using SQL Injection.