ISSN No. 0976-5697

Volume 5, No. 7, September-October 2014

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

A Review on Web Security and Its Applications

ShikhaVashist, Ayush Gupta
Student, Dronacharya College of Engineering,
Gurgaon, Hr., India-123506

Abstract: It has been observed for a quite long time that Web Security has been one of hot research areas, from point of view of be it either analysis or detection and later developing mitigation plans. Web security threats have undergone much sophistication compared to their initial introduction and they are becoming more & more evolved every day. The evolution might be in terms of new ways of attack or bringing in resistance to using [1]simulated OS or VM environments. Also, there has been considerable shift in the target of attacks in recent years. Earlier, clients were ignored while choosing targets. But, in recent years client user has become the main target for attacks as the adversary believe that the end user is the weakest link in the security chain. As a result of all these latest developments traditional security tools have been ineffective against these new attacks either for detecting or analysing the attacks. In this regard this paper presents a brief survey of research challenges and open issues in the area of web security under the suitable subtitles depending upon type of attack associated with the issues.

Keywords: web security, mitigation, evolution, resistance

I. INTRODUCTION

The increased usage of the Internet and network technology has changed the focus in assessing computer environment. The traditional approach considers the location of hardware and equipment first and then the data stored on the hardware. Also, these assessments were principally at specific points in time and principally[3]compliance-based reviews. With network-based technology (a.k.a. net centric technology), the primary concern is on the network and the contents of information. Also, an assessment of network technology is focused on the implementation and management of real-time controls to meet the business needs and to provide continuous scanning [2]. Security is not a one-time event. It is unsatisfactory to secure your code just once.[5]A secure coding initiative should deal with all stages of a program's lifecycle. Secure web applications are possible only when a secure SDLC is used. Secure programs are secured by design, during development.

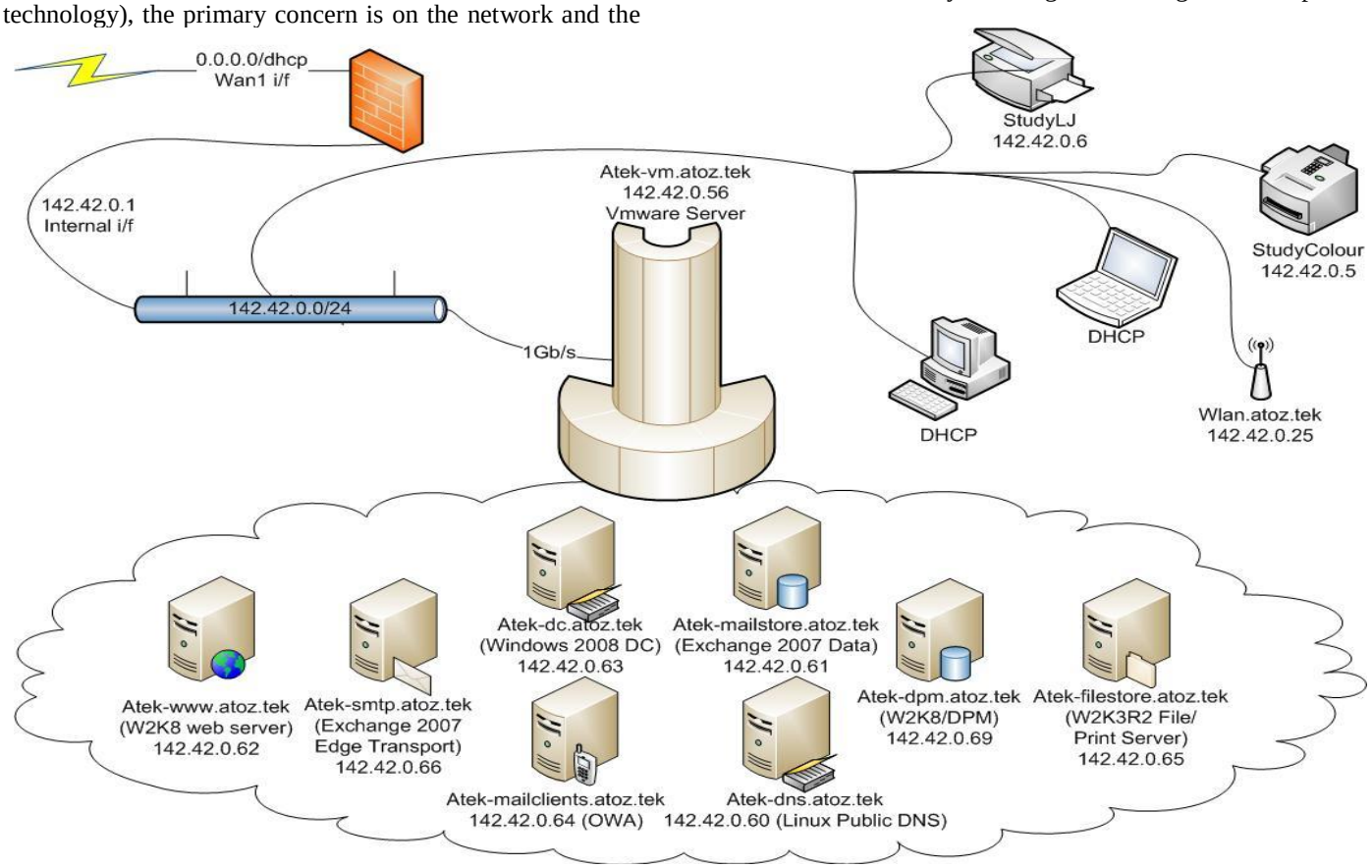

Figure: 1 Providing physical web security using IP address

A system and method for providing security for an Internet server. The systemcomprises: a logical security system forprocessing login and password datareceived from a client deviceserver session in order to authenticate a user; and a physical security system for processing Internet protocol (IP) addressinformation of the client device in order to authenticate the client device for the duration of the server session.

II. SYSTEM AND WEB SECURITY AGENT METHOD

Network security administrators are enabled with their customizable certificate authority reputation policy store which is informed by an independent certificate authority reputation server. The custom policy store overrides trusted root certificate stores accessible to an operating system web networking layer or to a third party browser. Importing revocation lists or operating system is made redundant. Proactive remediation is enabled to delete or disable root certificates in trusted operating system root certificate stores or in trusted browser root certificate stores by a web security agent installed at assigned endpoints. This removes the need for additional hardware or synchronous remote access over the protected endpoints.

III. WEB SECURITY VIA RESPONSE INJECTION

System and methods for injecting content into a response for improving client-side security. The system has a content injection service external to network edges of at least one system. The content injection service receives a request from a client within the at least one system and identifies or anticipates a potential threat associated with the response. The content injection service is configured to determine an appropriate counter for the identified or anticipated potential threat and in response injects content into the response according to the potential or anticipated threat identified.

IV. PRECISE WEB SECURITY ALERT

A method for providing an alert when a potentially or likely malicious web site is browsed to by a user. The method maintains web site identification details. If a web site purporting to be a known, previously identified, encountered and utilized web site is browsed to and requests information, the user is alerted to the precise differences between the stored web site historical identity and the identity of the present requester.

V. WEB HOSTED SECURITY SYSTEM COMMUNICATION

A distributed proxy server system is operable to receive a request for Internet data from a user, obtain the user identity, store at least one cookie on the user's web browser identifying the user, and filter undesired content before forwarding requested Internet data to the user. [7]A master cookie is associated with the proxy server including user identity information, and an injected domain cookie is associated with the domain of the requested Internet data including user identity information.

VI. ENFORCING WEB SECURITY THROUGH USER SPECIFIC XML SCHEMAS

A method of enforcing web security, by:

a. Receiving an incoming request
b. Applying a plurality of [2]XML customized schemas to the incoming request
c. Simultaneously validating the incoming request and determining whether the incoming request is authorized; and then,
d. (i) processing the incoming request if the incoming request is both valid and authorized, (ii) sending the incoming request to an authenticator if the incoming request is valid but not authorized, or (iii) ceasing operation on the incoming request if the incoming request is not valid.

VII. WEB SECURITY FLAW DETECTION METHOD

The invention relates to a security test of WEB application, and wants to provide a WEB dynamic security flaw detection method based on JAVA. The [6]WEB dynamic security flaw detection method based on JAVA is used for detecting the security flaws of a WEB application system, and comprises the following steps: modifying JAVA middleware; performing fuzzing test and dynamic flaw tracking. Due to the acceptance of the WEB dynamic security flaw detection method, more WEB security flaw problems can be found very fast, the security flaw range of black box test can be covered better, more deep WEB security problems can be picked, the problem of high cost in white box test can be solved, the specific position of a flaw code can be known more accurately, and lower missing report rate and error report rate in a detection are ensured.

VIII. CONCLUSION

Web applications reach out to a larger but less-trusted user base than legacy client-server applications. Yet they are more vulnerable to attacks. Various companies are starting to take initiatives to prevent these types of break-ins. [4]Code reviews, extensive penetration testing, and intrusion detection systems are some few ways that companies are battling a growing problem. Badly, most of the solutions available nowdays are using negative security logic (working with a list of attacks and trying to prevent against them). Negative security logic solutions can stop known, generalizedattacks, but are ineffective against the kind of targeted, malicious hacker activity outlined in the paper. If there is a consistent message among these attacks, the key to mitigate these vulnerabilities is to sanitize user input before processing it.

IX. REFERENCES

[1]. OWASP Top 10 Web Application Vulnerabilities.http://www.applicure.com/blog/owasp-top-10-2010

[2]. E. Chien. Malicious Yahooligans. http://www.symantec.com/avcenter/reference/malicious. yahooligans.pdf, 2006.

[3]. S. Di Paola. Wisec security. http://www.wisec.it/sectou.php?id=44c7949f6de03, 2006.

[4]. Ú. Erlingsson and F. B. Schneider. IRM enforcement of Java stack inspection. In Proc. IEEE Security and Privacy, 2000.

[5]. O. Hallaraker and G. Vigna.Detecting malicious JavaScript code in Mozilla.In Proc. IEEE Conf. on Engineering of Complex Computer Systems, 2005.

[6]. B. Hoffman. Ajax security. http://www.spidynamics.com/assets/documents/AJAXdangers.pdf, 2006.

[7]. T. Jim, N. Swamy, and M. Hicks. Defeating script injection attacks with browser-enforced embedded policies. In WWW, 2007.