Attacks in WEB Based Embedded Applications

Yaashuwanth .C, Research scholar
Dept. of Electrical and Electronics Engineering,
Anna University Chennai, Chennai 600 025, Tamilnadu, India.

Dr. R. Ramesh, Assistant Professor,
Dept. of Electrical and Electronics Engineering,
Anna University Chennai, Chennai 600 025, Tamilnadu, India.

Abstract

This paper deals with the issues related to embedded applications when they are implemented in internet. There are various attacks in embedded systems when implemented in the internet. These attacks have a negligible effect in the operating system which results in the decrease in the system performance. But in embedded system case, it has life and death consequence attached to it. Many of these embedded systems work in hazardous environment where a system failure results to catastrophic effects. Here a study of various attacks are discussed and a new architecture has been proposed to secure Web based embedded systems from the attacks.

Keywords*: Embedded systems, Web, Internet, attacks, secured layer*

1. Introduction

There are a number of security algorithms which prevent attacks when Web services are implemented. In general purpose operating systems, there are hardly a handful of algorithm which prevents Web attacks in Embedded systems. This paper gives a general study of various attacks that happens in Web based embedded systems.

Dong Haung [1] proposed a new ontology for representing security constraints as policy and a semantic policy framework for the management of the policies. The growth of internet has accompanied the growth of e services which resulted in increasing attacks on them by malicious individuals. The authors [4] highlighted the need of security. The conceptions about security of Web services and Degree of Safety for Web Services (WS-Dos), and the duration of Web service execution time, are introduced in the paper. In addition, a securing logical hierarchical structure for Web Service application based on an extended Web Services security architecture model with five elemental objects, such as resources, services, roles, protocols and methods object is analyzed Moreover, an integrated security solution has been developed and the results of application showed that the solution builds a confidence and authentication security environment for all roles in the process of dynamic B2B trade [8]. The authors [5] sketched the MAWeS architecture, illustrating how to use it to optimize the performance of a typical compound Web Services application while at the same time guaranteeing that a set of security requirements, expressed by a security policy, are met.Chen el al [6] aimed at clarifying security concern by conducting a quantitative performance evaluation of WSS overhead. Based on the evaluation, an extension of the existing Web services performance model is made by taking the extra WSS that overheads into account. The extended performance model is validated on different environments with different messages sizes and WSS security policies. Micheal lesk et al [7] explained Web Service security in a federated environment that describes how the Web services are implemented and are secured in a Web environment

Kevin [2] described the objective of improving Internet messaging by redesigning it as a family of Web services, an approach that is call WSEmail. This paper illustrated architecture and describe some applications. Since increased flexibility often mitigates against security and performance, here the steps for proving security properties and measuring the performance of the system with its security operations are focussed. The authors proposed [3] an agent based policy aware framework for Web Services security. In this framework, a policy language called ReiT which is a declarative language based on the rules and ontology is introduced. The non-structural knowledge is represented by rules and the structural temporal knowledge is represented by ontology Moreover, the authors propose a mixed reasoning mechanism to evaluate the ReiT policy. The access control policy including the context of the user and Web Services is evaluated by the reason or in addition, policy aware BOID agent to authorize the access control of the Web Service is presented. And the authors implemented the policy aware of BOID agent by extending the JADE platform.

2. Study of various attacks in Real Time Embedded Systems

2.1 Denial of service and distributed denial of service attacks.

A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the concerted efforts of a person or people to prevent an Internet site or service from functioning efficiently or at all, temporarily or indefinitely. Propagators of DoS attacks typically target sites or services hosted on high-profile

Web servers such as banks, credit card payment gateways, and even root nameservers. The term is generally used with regards to computer networks, but is not limited to this field, For example, it is also used in reference to CPU resource management.

One common method of attack involves saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable. In general terms, DoS attacks are implemented by either forcing the targeted computer(s) to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.One step ahead, DDos does is capable of doing more harm. With this, attacker can use the victims system to infect other connected systems or send a spam. Attacker can find a weakness in the system and can inject a malware or software which can be remotely used by using this, now attacker can make the server “a slave” and send spams or get access to files using its permission. Thousands of system can be targeted from a single point.

DOS and DDOS can also happen in embedded systems since the malicious hacker can gain access to the embedded Web server and use all the server resources such as limited bandwidth which in turn leads to denial of service for legitimate embedded client from accessing the service

2.2 Threat from Key Logging.

Keystroke logging (often called keylogging) is an action of tracking (or logging) the keys struck on a keyboard, typically in a covert manner so that the person using the keyboard is unaware that their actions are being monitored. There are numerous keylogging methods, ranging from hardware and software-based approaches to electromagnetic and acoustic analysis.

The users in the embedded Web client must ensure that their keystroke logging must not be tracked by an imposter. The snooper can gain entry if he is able to track the key logging of the end user.

2.3 IP Spoofing.

IP address spoofing or IP spoofing refers to the creation of Internet Protocol (IP) packets with a forged source IP address, called spoofing, with the purpose of concealing the identity of the sender or impersonating another computing system or an embedded device.

2.4 Buffer Overflow

A buffer overflow, or buffer overrun, is an anomaly where a program, while writing data to a buffer, overruns the buffer's boundary and overwrites adjacent memory. This may result in erratic program behavior, including memory access errors, incorrect results, program termination (a crash), or a breach of system security. Buffer overflows can be triggered by inputs that are designed to execute code, or alter the way the program operates. They are thus the basis of many software vulnerabilities and can be maliciously exploited. Bounds checking can prevent buffer overflows. Programming languages commonly associated with buffer overflows include C and C++, which provide no built-in protection against accessing or overwriting data in any part of memory and do not automatically check that data written to an array which (the built-in buffer type) is within the boundaries of that array

In embedded system this attack poses a greater threat since embedded systems will only allocate a small amount of memory. So any small type of this attack may lead to overflow of buffer at end user side and hence the system will crash

2.5 Format String Attack.

Format string attacks are a class of software vulnerability. Format string attacks can be used to crash a program or to execute harmful code. The problem stems from the use of unfiltered user input as the format string parameter in certain C functions that perform formatting, such as `printf()`. A malicious user may use the `%s` and `%x` format tokens, among others, to print data from the stack or possibly other locations in memory. One may also write arbitrary data to arbitrary locations using the `%n` format token, which commands `printf()` and similar functions to write the number of bytes formatted to an address stored on the stack This type of attack is also common in embedded systems which causes the embedded systems to crash

2.6 SQL Injection Attack.

SQL injection is a code injection technique that exploits a security vulnerability occurring in the database layer of an application. The vulnerability is present when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and thereby unexpectedly executed. It is an instance of a more general class of vulnerabilities that can occur whenever one programs or scripts the language that is embedded inside another. SQL injection attacks are also known as SQL insertion attacks.

2.7 Cross site scripting.

Cross-site scripting (XSS) is a type of computer security vulnerability typically found in Web applications that enables malicious attackers to inject client-side script into Web pages viewed by other users. An exploited cross-site scripting vulnerability can be used by attackers to bypass access controls such as the same origin policy. Cross-site scripting carried out on Websites were roughly 80% of all security vulnerabilities documented by Symantec as in

2007. Their impact may range from a petty nuisance to a significant security risk, depending on the sensitivity of the data handled by the vulnerable site, and the nature of any security mitigations implemented by the site's owner.

2.8 Virus and worms.

A virus is a program that can copy itself and infect a computer or any embedded device. The term "virus" is also commonly but erroneously used to refer to other types of malware, including but not limited to adware and spyware programs that do not have the reproductive ability. A true virus can spread from one computer to another (in some form of executable code) when its host is taken to the target computer; for instance a user can sent it over a network or the Internet, or carried it on a removable medium such as a floppy disk, CD, DVD, or USB drive. Viruses can increase their chances of spreading to other computers by infecting files on a network file system or a file system that is accessed by another computer.

As stated above, the term "virus" is sometimes used as a 'catch-all' phrase to include all types of malware, even those that do not have the reproductive ability. Malware includes computer viruses, computer worms, Trojan horses, most rootkits, spyware, dishonest adware and other malicious and unwanted software, including true viruses. Viruses are sometimes confused with worms and Trojan horses, which are technically different. A worm can exploit security vulnerabilities to spread itself automatically to other computers through networks, while a Trojan horse is a program that appears harmless but hides malicious functions. Worms and Trojan horses, like viruses, may harm a computer system data or performance. Some viruses and other malware have symptoms noticeable to the computer user, but many are surreptitious or simply do nothing to call attention to themselves. Some viruses do nothing beyond reproducing themselves.

2.9 Password cracking

Password cracking is the process of recovering passwords from data that has been stored in or transmitted by a computer system and embedded client systems. A common approach is to repeatedly try guesses for the password. The purpose of password cracking might be to help a user recover a forgotten password (though installing an entirely new password is less of a security risk, but involves system administration privileges), to gain unauthorized access to a system, or as a preventive measure by system administrators to check for easily crackable passwords.

3. Proposed Architecture

When real time applications are implemented with Web services they are subjected to various attacks which has been discussed. So there is a need to develop a proposed architecture to counter these attacks.

The proposed architecture solves the drawbacks of all the issues related to Web based embedded systems. The proposed architecture identifies the source of attacks. This is accomplished by deploying our defense systems in our distributed routers in order to examine incoming messages and place the headers. By this way, we can examine the message header and traceback the originator and prevent further attacks.

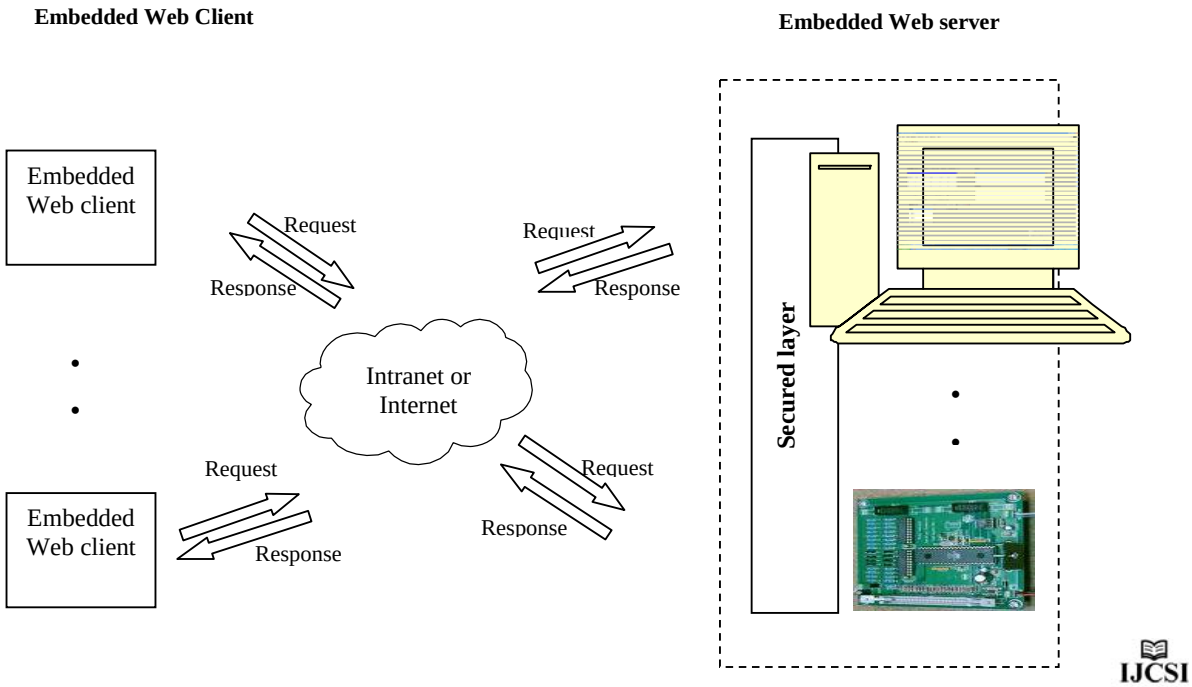

Fig. 3.1 Proposed architecture

4. Conclusion

This paper finally tune all the attacks related to embedded systems these attacks are related to denial of service to the Web client. Thus also a proposed architecture which is developed and implemented to prevent these type of attacks.

References

[1] Dong Huang "Scematic Description of Web Service Security Constraints" Proceedings of the Second IEEE International Symposium on Service-Oriented System Engineering IEEE2006. http://WWW.ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4027101

[2] Kevin D. Lux, Michael J. May, Nayan L. Bhattad, and Carl A. Gunter"WSEmail: Secure Internet Messaging Based on Web Services" Proceedings of the IEEE International Conference on Web Services IEEE2005. http://WWWieeexplore.ieee.org/iel5/10245/32665/01530785.pdf?arnumber=1530785

[3] Jlan-xin li,Bin li,Liang li and Tong sheng che "An Agent-based Policy Aware Framework for Web Services Security" IFIP International Conference on Network and Parallel Computing – Workshops IEEE 2007 http://WWWieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4351593

[3] George Yee and Larry Korba "Negotiated Security Policies for E-Services and Web Services" Proceedings of the IEEE International Conference on Web Services (I IEEE2005. http://WWWieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1530852

[4] Massimiliano Rak, Valentina Casola, Nicola Mazzoccca Emilio Pasquale Mancini and Umberto Villano ' Optimizing Secure Web Services with MAWeS: a Case Study". http://WWWieeexplore.ieee.org/iel5/4543879/4550291/04550321.pdf

[5] Shiping Chen1, John Zic, Kezhe Tang, and David Lev "Performance Evaluation and Modeling of Web Services Security" International Conference on Web Services IEEE2007. http://WWWieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4279628

[6] Michael Lesk, Martin R. Stytz and Roland L. Trope " providing Web service security in a federated environment" International conference on Privacy and Security IEEE 2007. http://WWWieeexplore.ieee.org/iel5/8013/4085579/04085599.pdf

[7] Yuan wo,, Bo-qin feng', Jin-Cang and Zun-Chao li "SX-RSRPM: A Security Integrated Model for Web Services" Proceedings of the Third International Conference on Machine Learning and Cybernetics IEEE 2004. http://WWWieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1378538

About the authors

Mr. C.Yaashuwanth completed his B.Tech. degree in Information technology at BSA CRESCENT Engineering College Chennai, He completed M.E. in Embedded System Technologies at VEL TECH Engineering college Chennai. He is currently Pursuing his Ph.D program under Dr. R .Ramesh

Dr. R. Ramesh pursued his B.E. degree in Electrical and Electronics Engineering at University of Madras, Chennai, and completed his M.E. degree in Power Systems Engineering at Annamalai University, Chidambaram. He received his Ph.D. degree from Anna University, Chennai and has been a faculty of Electrical and Electronics Engineering Department of College of Engineering, Guindy, Anna University, Chennai since 2003. His area of interest are Real-time Distributed Embedded Control On-line power system analysis and Web services.