Free essay on software security assessment

Government, Corruption

Introduction

Software transports instructions to the computer that it follows to operate various functions. Until now, it is not perfectly secure enough to be protected against hackers, attackers, and undesired users etc. Nothing can be done for software to be put in a shell that is impossible to be affected ever. Due to the facts, software security assessment is essential to be taken place so that no effort remains for it is accessible to attackers. Software security assessment helps and guide to identify and prevent software vulnerabilities that harm data and information stored.

Software Vulnerability

Software vulnerability can be described as a weakness through which attacker grasp all the information stored in software or the information is accessible to them. That is the reason that in order to have quality software it is designing, development, and deployment should be done by experts with the practice of quality control planning and testing at each stage from designing to the implementation of software (Cetnerowski, 2011). If these issues are not given importance then it costs a lot in terms of customer’s trust and confidence, loss of information that is captured by undesirable attackers, damage of reputation faced by software producers etc.

Memory Corruption and Buffer Overflow

Software is often unintentionally gets insecure when the computer programs start to encounter errors in the contents. This is called memory corruption and if contents with errors and flaws remain to be used in computer then it eventually results in a crash of program. Furthermore, while memory is

corrupted then it can lead to programs behaving very unpredictably too. Memory corruption has many reasons as sometimes memory in program is utilized more than its capacity. The issue is called buffer overflow, which is extremely important to be considered. Buffer overflow is a flaw of programming that comes to the computer through various viruses, affecting security of software.

C languages and Pointer Arithmetic

Shell Code and Software Security Assessment

In a software security, shell code plays an important part. Shell code is a set of codes used negatively by attackers, having two types local and remote. It gives attackers access to the data and information stored in computers and their systems. By using a shell code attacker can try carious options of codes and the one that matches gives them access to confidential areas, systems, programs, and information. Software security and assessment is also important when it comes to data storage. Computer and its systems comprise very confidential and important data and information that need to be protected from all aspects.

Protection Mechanism

Protection Mechanism is a mechanism in computer sciences that assist in enforcement of security policies (Wang et. al, 2009). They are directly exposed to the threats and content to the system. Such mechanisms support the architecture to ensure security policies in a computer system. They are responsible for ensuring that relevant information is exposed to relevant people. Protection mechanism includes perception defense, structural

defense, content control, and behavioral mechanism to conduct its activities of support in a system.

Assessing Memory Corruption Impact

Occurrence of memory corruption in a computer program takes place when content inside a memory location are modified without intent leading to programming errors. Whenever corrupted memory is utilized again, it results in programming crash or even uncanny behavior from the programming. However, the threat for such issues can be countered by pointer arithmetic and explicit memory management.

C Language Issues

C language even though is a sound programming language but despite that, it has some issues that can cause problems in the programming. Some among the issues include non-terminated comments, accidental assignments, accidental Booleans, unhygienic macros, mismatched header files etc. These issues can cause interruptions in programming.

Data Storage Overview

In a computer system, data storage comprises of components related to recording media and is utilized to retain data in a digital format. These components are said to be the crux of a computer system and are essential. Expensive and small storage are often kept near the CPU which manipulates the data and slower, inexpensive, and larger away from it.

Arithmetic Boundary

Arithmetic boundary is the minimum and maximum possible values that are determined by their basic representation in the memory in C’s basic integer

(Haj Said, 2011). Whenever the maximum boundary limit value is passed, it is a numeric overflow condition. Similarly, when a value generated is lower than the minimum boundary limit the condition that occur is numeric underflow condition.

Type Conversion

Type Conversions are ways of changing data type from one entity to another either explicitly or implicitly. The major reason behind that is to gain an advantage over type representations. However, one evident factor in this conversion is that it varies in accordance to the programming language, as for every programming language there are different rules for conversion.

Operators

Operators are procedures to produce an output where an input is needed. It can be described as a representative to a certain action for instance the + sign in mathematics or in computers mean addition. In programming languages, they are used in assisting testing conditions.

C Language Nuances

C language despite being a sound language in computer programming has certain drawbacks. This include lack of guarantee in order evaluation, structure padding, precedence of complex expressions, security problems related to macros/preprocessors, and typos. These problems make C language vulnerable to threats.

References

Cetnerowski, A. (2011). The art of software security assessment: Identifying and avoiding software vulnerabilities. Software Quality Professional, 14(1),

48.

Wang, Y. Y., Lively, W. M., & Simmons, D. B. (2009). Software security analysis and assessment model for the web-based applications. Journal Of Computational Methods In Sciences & Engineering, 9179-S189.

Haj Said, F. (2011). Security-based risk assessment for software architecture. (Order No. 3530557, West Virginia University). ProQuest Dissertations and Theses, , 127