Vulnerabilities in software Security

By Group Two Team Five (Training and Recruitment)

Abstract

In this report we be discussing the different types of vulnerabilities that can be found in software.

Kerri-Anne O’Brien 30014123

Syed Zaki Zaidi 30010702

Saleh Mohamed 30001162

Hossam Farfour 30008722

Luke Waddington 18030327

**Introduction**

This section of the report will discuss the various vulnerabilities in cyber security. A vulnerability in a computer system is a potential weakness in a computer system or computer network in which could be used to inflict severe damage to the system or enable a hacker or malicious software to manipulate the computer system. It “is a flaw or weakness in a system or network that could be exploited to cause damages, or allow an attacker to manipulate the system in some way” (Dosal ,2020).

Computer Systems have a number of potential vulnerabilities including: Malware- This includes things like **Ransomware, Trojans and Worms.** Ransomware revolves around malicious software which is used to encrypt the data storage drives of its potential victims, thus leaving them completely inoperable to the owner. From this, the encryption key is only returned in exchange for payment. Trojans act as a form of delivery for malware. Trojans conceal themselves as looking like legitimate software in an attempt to fool the user into installing it onto their computer system. This malware can cause serious damage due to the fact that they can avoid being detected by anti-virus security systems as they appear as legitimate software, but when opened, they can release severe viruses into the computer system. Worms are a type of software which can duplicate itself and spread to computer systems through a variety of different methods.

There are a number of factors which can leader to vulnerabilities in a computer system. For example, these can include things such as bugs, weak password, Buffer overflow, the use of unrestricted upload of dangerous file types, cross-site scripting and forgery and the use of broken algorithms. Moreover, there are five parts of system securities which can be negatively impacted as a result of computer security vulnerabilities. This can include the confidentiality of a computer system, which revolves around keeping the user’s data safe from being obtained by third parties. Usability, which guarantees the satisfaction users will have from using various services and packages that are provided from computer systems. As well as this, Reliability which revolves around the efficiency of a computer system (Comodo, 2018)*.*

*-Saleh Mohamed 30001162*

**Password vulnerabilities**

There are many ways a password of a user can be vulnerable to an attacker but three of the most common are as follow:

1. Phishing

Phishing is a type of a cyber-attack which done to get all the private and confidential information of the user which include bank details and data by hacking into the account of the user it is done through emails or text messages where the hacker sends a link which looks irresistible and asks the user to click on it. Once the link is clicked the hacker gains the access of users account and changes the password. There is a way phishing can be avoided it is by looking at the URL of the of the websites which are sent there will be a major difference in both (Hoffman,2019).

1. Social engineering

Social engineering is an act of exploiting human nature and making them reveal all their confidential information such as passwords. As (Symanovich, 2011) describes it the whole concept behind social engineering is to play with the feelings of the victim. Social engineers play with human emotions with in such a way that out of trepidation they end up giving their valuable information. A user can save himself from these social engineers by not replying to any suspicious emails or calls.

1. Brute force attack

Brute force attack is a type of attack when the hacker tries hundreds of passwords through a software until it lets him in to the victims account. The way a person can prevent himself from this attack is by having complex password of minimum 12 characters with numbers and symbol included.

*-zaki 30010702*

**SQL Injection**

SQL injection is one of the major cyber threats in which attackers use SQL commands to get into a website or a database and get all confidential and private information of the user or an enterprise. As stated in Kaspersky (2021) it one of the most dangerous cyber threat because it can be used against any website which use Structured Query Language as its data base.

There are many a hacker can make SQL injection for his favour but some of the examples are as follow:

* Retrieving hidden information:

A hacker can insert an SQL commands into a website and make all unavailable product available to a user which user could not see before by just writing few lines of code and by playing with the where command.

* Accessing accounts without passwords:

The attacker can eject SQL injection to websites URL and make the password option concealed meaning by just entering the username the webpage will let the hacker into the account through which hacker can gain all private and data and information

Majority of the SQL injections occurs because of the where clause of a select query. There are many ways from preventing from SQL injections which are using software which detect any breaking that could happen also SQL injections can be prevented by using parameter queries instead of concatenating them also as stated by Acunetix (2021) we should never trust user input this reduces major risk of an SQL injection

*-zaki 30010702*

**Buffer Overflow**

A buffer overflow can occur inadvertently, but it can also be caused by a malicious actor sending carefully crafted input to a program that then attempts to store the input in a buffer that isn't large enough for that input.

Other buffer-related attacks include integer overflow, which is when a number is used in an operation, the result of which requires more memory to store. Buffer overflows are one of the oldest and most common causes for arbitrary code execution vulnerabilities, and applications written in programming languages like C and C++ are more prone to such coding mistakes than other languages.

Buffer overflows typically have a high severity ranking because they can lead to unauthorized code execution in cases where attackers can control the overwritten memory space outside the targeted buffer and can redirect a function pointer to their malicious code.

* Hossam Farfour 30008722

**Bugs**

When it comes to computer science, a bug in a computer system is an error that can be found in the source code which leads to programs malfunctioning and crashing unexpectedly. These bugs can greatly impact a computer systems performance and its efficiency. There are a number of different types of bugs in computer systems.

Some of these bugs include Logic bugs, these bugs result in errors in functions that use scripts. Another is Runtime bugs. These bugs lead to errors being caused when the program is in the process of running. These cause runtime errors that come about in the event that a program attempts to call a function which is non-existent. Moreover, arithmetic bugs are bugs that occur as a result of integer overflow or integer underflow and can result in an arithmetic operation that leads to a numeric value beyond that of the given range. Finally, there are interfacing bugs. These occur in the event that an incompatible application programming interface protocol, protocol implementation, hardware handling or other systems are connected to a computer (*Techslang*, no date).

The difference between a bug and an error is that an error message is something that occurs when something goes wrong in a program whereas a bug is the cause of the error and is an issue in the actual code. Some bugs, such as software bugs can be fixed through the activation of software patches whilst hardware bugs can be removed by firmware updates as well as certain drivers (*ComputerHope*, 2020).

*-Saleh Mohamed 30001162*

**Missing Authorisation**

Authorisation is a process where the application determines what resources the user should access once they have been authenticated. (Fraser, 2021)

Authorisation is a necessary tool as it limits the resources that the user can access, such as other user’s files. Renowned methods of authorisation include:

* Token-based – Users are given a “ticket” which states what they have access to, once the user signs out, the token becomes invalid. Tokens provide a secondary layer of security over traditional password or server-based methods as the administrator has extensive control over what the user can do. (Gan, Chen, Zhou, & Zhu, 2021)
* Role-Based Access Control (RBAC) – Users are identified in a role that is pre-defined and stipulates the permissions they have; access can also be prohibited via user IDs (Ferraiolo & Kuhn, 1992).
* Access Control Lists (ACL) – ACLs specify which users have access to the associated files/folders, user IDs or roles can be assigned for exclusive access. (Shirey, 2021)

A majority of modern, multi-user operating systems use RBAC, from this, administrators can incorporate the “principle of least privilege”, where users should only be authorised what is deemed essential to their functions to optimise security.

*-Luke Waddington 18030327*

**Missing Authentication**

Authentication is the verification of a person or system through an identification method. Once authentication is completed certain types of authorisation can be given based on who is requesting access. Missing authentication makes it easier for hackers to gain access to software and where-ever possible “security considerations should be employed to make the job of attackers much more difficult” (McClure, Scambray, Kurtz, 2001 pp.18).

To increase security and help prevent against attacks all software should have good authentication methods in place. The best way to prevent hackers from gaining easy access to your software is to have one or more types of authentication built into it. The most common form of authentication is a password or passcode and “is one of the most simple and effective approaches for authentication in a client/server environment” (Xu, Zhu, Feng, 2005). Other methods can be used such as biometrics e.g., fingerprint id, face id; physical methods such as inferred door locks and social methods such as visitor badges.

For software I would go with encryption and passwords. The principles of data encryption are to make data hard to read and therefore intercept. As described by Bartlett (2014, p.78) “Encryption is the art and science of keeping things secret from people you don’t want to know them, while revealing them to those you do”. There are four main ways in which data can be encrypted to make it more secure. These are: asymmetric and symmetric encryption, random number generator and the hash function.

* *Kerri-Anne O’Brien 30014123*

**Sniffing**

Sniffing is a type of attack used by hackers to gain sensitive information. Packet sniffing as defined by (Ince, 2019) “can be a legal activity which involves system staff checking a network, or a criminal activity aimed at gaining unauthorized access to a network.” Illegal packet sniffing is often done over a network where two people share information, and the hacker listens in. The hacker will install a program onto the targets device which will watch everything they do. The type of data they usually steal is personal information about a person such as a name, date of birth and address for purposes such as identity fraud. They could also steal bank details, usernames, and passwords for stealing money.

Content sniffing, as (Barua, Shahriar and Zulkernine, 2011) describes is a “starting point” which makes victims susceptible to other types of attacks and sniffing attacks should be spotted as early as possible to combat this.

Although some sniffers are used for malicious purposes this was not their intent “due to malicious users it has had negative connotations in the past” and there are many sniffers that are used for good purposes such as admin sniffers that test systems (Orebaugh, 2004, pp.2).

The best way to prevent hackers from installing sniffers is to have up to date browsers and secure encryption across them.

* *Kerri-Anne O’Brien 30014123*

**Conclusion -**

In concluding vulnerabilities in software security causes breach of privacy and valuable data loss which causes billions of dollars of loss every year. There are many ways how a software’s vulnerabilities can be exploited by a hacker it can either be done after a user being tricked through phishing where user clicks on a suspicious link because of trepidation or download content from untrusted sources which leads malware and viruses such as trojan and worms into the computer of the user. Missing authentication also plays a big part as not having 2 factor logins makes it easier to gain access of one’s account. Sniffing makes a user vulnerable through network connections as all of its private and confidential data and information could be passed on through it. To save from all these kinds of vulnerabilities a user must have and updated anti-virus and firewall all the time and should never click on links or URL which seems suspicious and have 2 factor logins.

*-zaki 30010702*

Citations

Acunetix (2021). *What is SQL injection (SQLi) and how to prevent it*. Available at: <https://www.acunetix.com/websitesecurity/sql-injection/> (Accessed: 24 February 2021)

Bartlett, J. (2014). *The Dark Net*. London: William Heinemann - Random House. p78.

Barua, A, Shahriar, H, and Zulkernine, M. *Server Side Detection of Content Sniffing Attacks*. 2011 IEEE 22nd International Symposium on Software Reliability Engineering (2011): 20-29. Web.

Books24x7, Inc, and Syngress Media, Inc. *Ethereal Packet Sniffing*. First ed. Vol. V. 1. Rockland: Syngress, 2004. Jay Beale's Open Source Security Ser. Web.

Comodo(2018) *Computer Vulnerability: Definition.* Available at:

<https://enterprise.comodo.com/blog/computer-vulnerability-definition/>

(Accessed: 18, February 2021)

*ComputerHope* (2020) *Bug.* Available at:

[What is a Bug? (computerhope.com)](https://www.computerhope.com/jargon/b/bug.htm)

(Accessed: 19, February 2021)

Darrel Ince. *Packet Sniffing*. A Dictionary of the Internet (2019): A Dictionary of the Internet, 2019 10-24. Web.

Dosal, E.(2020) *Top 5 Computer Vulnerabilities.* Available at: <https://www.compuquip.com/blog/computer-security-vulnerabilities>

(Accessed: 20, February 2021)

Ferraiolo, D., & Kuhn, D. (1992). *Role-Based Access Control.* National Computer Security Conference.

Fraser, B. (2021, February 24). *RFC 2196 - Site Security Handbook*. Retrieved from Internet Engineering Task Force: https://tools.ietf.org/html/rfc2196

Gan, G., Chen, E., Zhou, Z., & Zhu, Y. (2021, February 24). *Token-Based Access Control - IEEE Journals & Magazine*. Retrieved from Institute of Electrical and Electronics Engineers: <https://ieeexplore.ieee.org/document/9031334>

Hoffman, J. (2019) *Most common password vulnerabilities you should know about.* Availableat: <https://wisdomplexus.com/blogs/common-password-vulnerabilities/> (Accessed: 23 February 2021).

Kaspersky (2021). *What is SQL injection.* Available at: <https://www.kaspersky.co.uk/resource-center/definitions/sql-injection> (Accessed: 23 February 2021).

*Techslang* (no date) *What is a computer bug?* Available at:

<https://www.techslang.com/definition/what-is-a-computer-bug/>

(Accessed: 22, February 2021)

McClure, Stuart, Scambray, Joel and Kurtz, George. *Hacking Exposed: Network Security Secrets and Solutions*. California: Osborne/McGraw-Hill. 2001 p.18.Web.

Shirey, R. (2021, February 24). *RFC 4949 - Internet Security Glossary, Version 2*. Retrieved from Internet Engineering Task Force: https://tools.ietf.org/html/rfc4949

Symanovich,S. (2018) *Emerging Threats.* Available at: <https://us.norton.com/internetsecurity-emerging-threats-what-is-social-engineering.html> (Accessed: 22 February 2021).

Xu, Jing, Zhu, Wen-Tao, and Feng, Deng-Guo. *An Improved Smart Card Based Password Authentication Scheme with Provable Security*. Computer Standards and Interfaces 31.4 (2009): 723-28. Web.