

Software Vulnerabilities

30012104, 1703285, 17129001, 18066291, 30008859

LEAVE MANAGEMENT SYSTEM

Table of Contents

[Vulnerability – what is. – Michal 30012104 1](#_Toc63349606)

[Types. 1](#_Toc63349607)

[SQL Injection – Michal 30012104 2](#_Toc63349608)

[Sniffing – Michal 30012104 3](#_Toc63349609)

**DDoS attack- WEI 17030285……………………………………………………………………………………………..4**

**Buffer overflow- WEI 17030285………………………………………………………………………………………..5**

**Reference list……………………………………………………………………………………………………………………6**

# Vulnerability – what is. – Michal 30012104

According to Wikipedia (Vulnerability (computing), 2021) vulnerability in computing is a weakness which can be exploit by attacker. Intention of attacker is to cross boundaries of privileges and perform unauthorized action within computer system. Vulnerability is also known as attack surface. Reason why is because threat actor must have at least one “tool” to connect him with system weakness point.

In the book “The art of Software Security Assessment” (Dowd, McDonald and Schuh, n.d.) we can find that software vulnerability is weakness in system that attacker can use to do something malicious, as examples author uses disturb and destroy a system, expose or alter sensitive information and take control of program or computer altogether.

## Types.

We will cover 7 different types of vulnerabilities:

1. SQL injection. – Michal 30012104
2. Sniffing – Michal 30012104
3. DDoS attack- WEI 17030285
4. Buffer overflow- WEI 17030285
5. Xx
6. Xx
7. Xx

# SQL Injection – Michal 30012104

According to portswigger.net SQL injection is a vulnerability that allows threat agent to interference with queries that application makes to its database. (Academy and injection, 2021) This can lead to attacker get data they would be normally would be able to retrieve. In many cases, an attacker can not only retrieve data but also alter or delete that data causing changes to application changes or behaviours. In bad scenarios attacker might even gain access to the underlying server or other back-end infrastructure.

“SQL injection usually occurs when you ask a user for input, like their username/userid, and instead of a name/id, the user gives you an SQL statement that you will unknowingly run on your database.” (SQL Injection, 2021).

Data sanitation is an absolute must when it comes to basic protection against SQL injections. We should simply pre-check all data before system entry. Removing all the garbage data allows us to make sure that the query won't break system and or database.

As we can read in (security.berkeley.edu, 2021) its still common misconception that input filtering is still seen as only measure programmers have to think of when designing the application. This is wrong as input filtering does not fix the underlying vulnerability. It is often good enough to stop the most trivial of the attacks.

To protect your application from SQL injection one of the most common practice is to parameterized statements, do not use dynamic SQL. When programming language like python talks to the database it uses database drivers. These simply allowing our application to build and run SQL statements and making requested changes to our database. Parameterized statements simply make sure that inputs passed into SQL are treated in a safe manner.

Configuring proper error reporting shouldn’t be skipped when designing the application. Database error messages should never be sent to a client web browser. This could lead to discovering by attacker technical details of our application and then be used to build successful malicious code to inject.

Needless to say that SQL injection is illegal and can have serious consequences such as massive fines and jail time. But the company who is being hacked merely fixing a problem is often only beginning of trouble. In many cases, companies suffer from bad PR and losing the reliability and trust of customers. For example one of the largest world credit card companies was hacked a couple of years back and some estimated claimed that over 100 million cards and more than 650 financial services companies were compromised. During prosecution, it was three corporate victims reported over 300 million US dollars of losses(Vaas, 2021). Not only large companies suffered due to this incident of SQL injection but also individuals through identity theft. Many of them had to deal with false charges associated with a stolen identity. The scale of this damage was described by the court as “immeasurable”. One of the attackers that pledge guilty was sentenced to 12 years of jail (US prosecutors launch largest-ever hacking fraud case, 2021).

# Sniffing – Michal 30012104

Sniffing in network traffic can be thought of as putting a bug on your telephone line, and listening to what two or more parties have to say. In networking, it will be watching and intercepting data packets that are being exchanged. When data transmitted isn’t encrypted in can be read using application design to intercept data called a sniffer. Using sniffer we can through FTP traffic, Web traffic, email traffic and get hold of sensitive information including passwords and ID’s, bank details etc. According to (Gregorczyk et al., 2020) sniffing is one of the most important techniques that attackers use to collect information. This information later can be used to exploit vulnerabilities within computer systems. Often packet sniffers just listen, display and log the traffic without making any modifications(Wikipedia, 2021). Gregorczyk points out that due passive nature of sniffing it can be very hard to detect.

As we can read in (Greycampus, 2021) There generally two types of sniffing :

* Active - When we sniffing in the switch, point that regulates the flow of data between ports. Active sniffing attacker floods switch with bogus requests. This leads to traffic to in all ports.
* Passive – this type of sniffing occurs in a hub. This type would operate on the data link layer of the network.

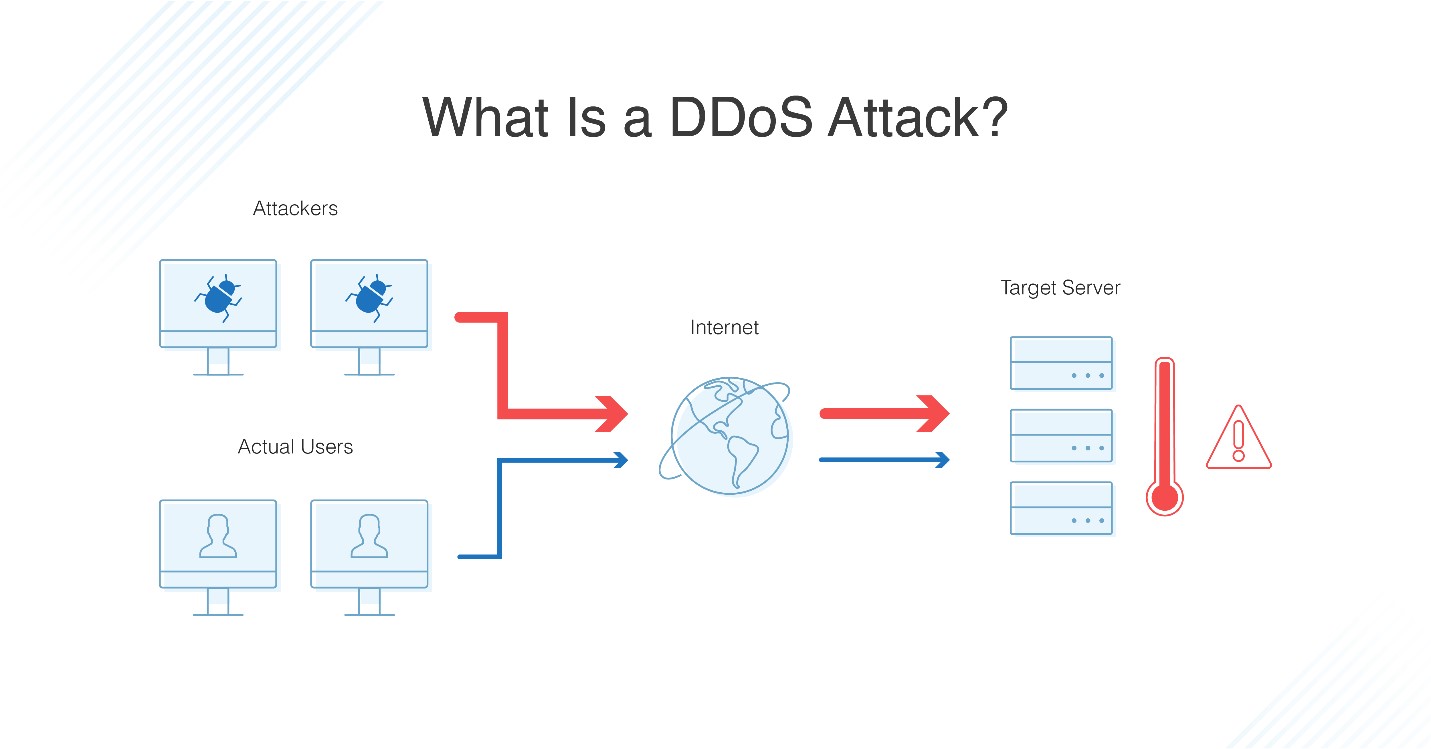
There are simple techniques that we should use when connecting the application to the network. These include avoiding all insecure protocols such as HTTP, FTP. Instead, we should always remember to connect application using protocols like HTTPS, SFTP, SSH. We should also consider encrypting our data whenever it is possible.

Other methods of sniffing prevention are using specially design tools to name few: Promqry, Sniffdet, Anti-sniffer.

According to The Register (Data-sniffing attack costs Heartland $12.6m, 2021) In 2009 Heartland Payments System corporation spend over 12milion dollars on encrypting credit card data, this was directly related to the security breach that had a place. Hackers using sniffing techniques got hold of cards holders sensitive data. Luckily these days end-to-end encryption are hugely popular and used in sorts of application (ex. WhatsApp).

DDoS attack – WEI 17030285

DDoS attack uses the tool that send a huge number of data packages to the target server then the target server will be coerced to accept the huge process take up many system resources and bandwidth. It is focus on TCP/IP protocol. DDoS always used to attack the bank and super company. If the user or manager of Web server do not know the reason of the close of website or service, it can also cause business credit problems. For these reasons we must learn how to stop or defence these attacks.



The DDoS attack have some frequently used methods.

1.Volumetric attacks: this is the most popular methods of DDoS attack. A program via the open port to send huge number of error permission to suppress the bandwidth. That means the legal request can not be process. The two main type of volumetric attack is UDP (user datagram protocol) floods and the ICMP (internet control message protocol) floods. (by the way I think we also can use this ideology to attack the DHCP server, use a huge number of fake IP address request to down the DHCP server).

2. Application layer attack: it is focus on the application layer of TCP/IP protocol or OSI. That means the target is the HTTP, HTTPS, DNS, SMTP, or some protocol in application layer. These attack always difficult to catch or debug because it will simulate the legal request easily.

3. Protocol attacks: it is always used to attack the networks that are used for verifying connection. They work by send purposefully slow or malformed pings so that the network uses up a lot of RAM try to verify these pings. It is always used to attack the firewalls.

How to defence these type of attack

Do not take it lightly, even you are a small company, you still have chance to be a target under the DDoS attacks. It is concerning the business reputation. So, you must have a plan to solve these problems.

1. Have a plan: firstly, you need an emergency response mechanism for DDoS. That means you need clearly check all your system and find the security loopholes. And debug if can not solve the bug immediately you should make plan to defence the attack.
2. Allocate roles: you should set a team for defence this type of attack in your IT department. And make sure all members can know the role of them and set the communication methods.
3. Install the defence tools: make sure you installed the defence tools; it is including but not limited to the firewall (hardware or software firewall), syslog server (collect the abnormal information), flow monitor.
4. Keep everything to update: all the system should be updated and always make backups. And make sure fixed the bug or loophole immediately.

Buffer overflow – WEI 17030285

The buffer overflow or buffer overrun is an abnormal that when the program needs to write data in the buffer, overruns the buffer’s boundary and overwrite adjacent memory locations.

Buffer area is used to store the data in a serial of memory address. Buffer overflow always caused by the error input. We can see the conventional calculator (it is always can calculate the fixed digits +-\*/) when we take over the digits, it will tell you an error such as this calculator can do 3 digits number you give it 999 add 1 then it will be lost one digit of number, then the data will be not correct.

Normally, the C/C++ is quite difficult to defence the buffer overflow. Because it can use the pointer to access memory just like assembly, so if you forget to write the defence code it will quite be easy to overrun the boundary of process or thread memory.

Buffer overflow attack via to write the exceed fixed digits data to make buffer overflow then destroy the heap and stack structure then make the program breakdown, it is reason about the programmer does not make the defence program to check the user input. For instance, void bufferOverflow()

{

Char a[4];

Char b[]={“copy char array”};

Strcpy(a,b);

}

We can clearly see that the A array have 4 char address and the b have 15 char address, so if we copy b to a then a will overflow to make this program breakdown.

So when we do the programming we must limit the user input in the buffer’s boundary to make sure the user input cannot destroy the system.

# Reference

En.wikipedia.org. 2021. *Vulnerability (Computing)*. [online] Available at: <https://en.wikipedia.org/wiki/Vulnerability\_(computing)> [Accessed 21 January 2021].

Dowd, M., McDonald, J. and Schuh, J., n.d. *The Art Of Software Security Assessment*.

Academy, W. and injection, S., 2021. *What Is SQL Injection? Tutorial & Examples | Web Security Academy*. [online] Portswigger.net. Available at: <https://portswigger.net/web-security/sql-injection> [Accessed 21 January 2021].

W3schools.com. 2021. *SQL Injection*. [online] Available at: <https://www.w3schools.com/sql/sql\_injection.asp> [Accessed 21 January 2021].

Vaas, L., 2021. *Hackers sentenced for SQL injections that cost $300 million*. [online] Naked Security. Available at: <https://nakedsecurity.sophos.com/2018/02/19/hackers-sentenced-for-sql-injections-that-cost-300-million/> [Accessed 4 February 2021].

BBC News. 2021. *US prosecutors launch largest ever hacking fraud case*. [online] Available at: <https://www.bbc.co.uk/news/technology-23448639> [Accessed 4 February 2021].

Security.berkeley.edu. 2021. *How to Protect Against SQL Injection Attacks | Information Security Office*. [online] Available at: <https://security.berkeley.edu/education-awareness/best-practices-how-tos/system-application-security/how-protect-against-sql> [Accessed 4 February 2021].

Gregorczyk, M., Zorawski, P., Nowakowski, P., Cabaj, K. and Mazurczyk, W., 2020. Sniffing Detection Based on Network Traffic Probing and Machine Learning. *IEEE Access*, 8, pp.149255-149269.

En.wikipedia.org. 2021. *Sniffing attack*. [online] Available at: <https://en.wikipedia.org/wiki/Sniffing\_attack> [Accessed 4 February 2021].

Greycampus.com. 2021. *Greycampus*. [online] Available at: <https://www.greycampus.com/opencampus/ethical-hacking/sniffing-and-its-types> [Accessed 4 February 2021].

Theregister.com. 2021. *Data-sniffing attack costs Heartland $12.6m*. [online] Available at: <https://www.theregister.com/2009/05/07/heartland\_breach\_costs/> [Accessed 4 February 2021].

# DNSstuff.com.2019.How to Stop, Prevent, and Protect Yourself from a DDoS Attack. [online] available at:< How to Stop and Prevent DDoS Attack - DNSstuff>[Accessed 4 February 2021].

En.wikipedia.org.2020. Buffer overflow. [online] Available at:< https://en.wikipedia.org/wiki/Buffer\_overflow#Choice\_of\_programming\_language>[Accessed 4 February 2021]