RAMA UNIVERSITY

www.ramauniversity.ac.in

FACULTY OF EGINEERING & TECHNOLOGY

MOBILE SECURITY

LECTURE -21

Umesh Kumar Gera
Assistant Professor
Computer Science & Engineering

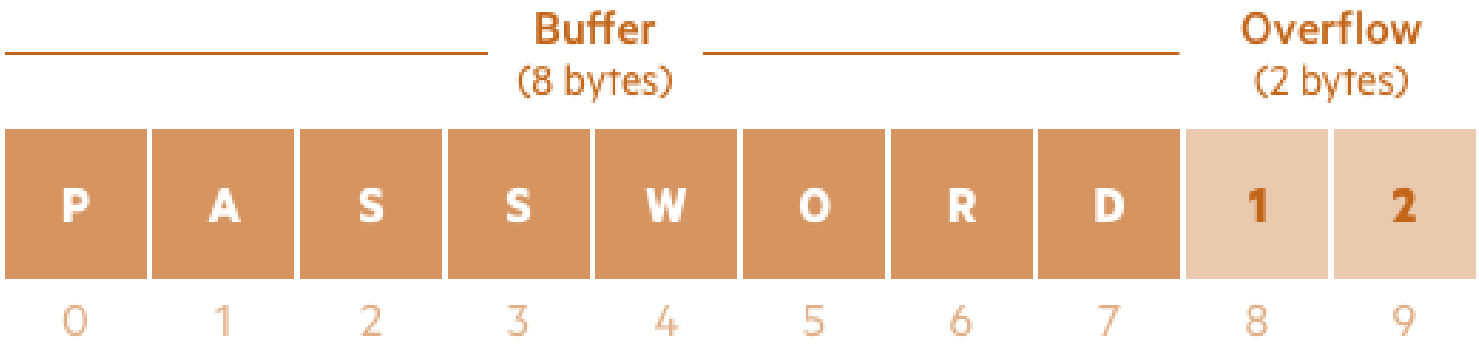
OUTLINE

- Buffer Overflow Attack
- What is Buffer Overflow?
- What is a Buffer Overflow Attack
- Types of Buffer Overflow Attacks
- How to Prevent Buffer Overflows
- PHP Security Best practices
- Security Issues in PHP CMS
- Possible solutions of PHP Security
- MCQ
- References

BUFFER OVERFLOW ATTACK

What is Buffer Overflow?

Buffers are memory storage regions that temporarily hold data while it is being transferred from one location to another. A buffer overflow (or buffer overrun) occurs when the volume of data exceeds the storage capacity of the memory buffer. As a result, the program attempting to write the data to the buffer overwrites adjacent memory locations.



BUFFER OVERFLOW ATTACK

What is a Buffer Overflow Attack

Developers can protect against buffer overflow vulnerabilities via security measures in their code, or by using languages that offer built-in protection.

In addition, modern operating systems have runtime protection. Three common protections are:

Address space randomization (ASLR)—randomly moves around the address space locations of data regions. Typically, buffer overflow attacks need to know the locality of executable code, and randomizing address spaces makes this virtually impossible.

Data execution prevention—flags certain areas of memory as non-executable or executable, which stops an attack from running code in a non-executable region.

Structured exception handler overwrite protection (SEHOP)—helps stop malicious code from attacking Structured Exception Handling (SEH), a built-in system for managing hardware and software exceptions. It thus prevents an attacker from being able to make use of the SEH overwrite exploitation technique. At a functional level, an SEH overwrite is achieved using a stack-based buffer overflow to overwrite an exception registration record, stored on a thread's stack.

BUFFER OVERFLOW ATTACK

How to Prevent Buffer Overflows?

Attackers exploit buffer overflow issues by overwriting the memory of an application. This changes the execution path of the program, triggering a response that damages files or exposes private information. For example, an attacker may introduce extra code, sending new instructions to the application to gain access to IT systems.

If attackers know the memory layout of a program, they can intentionally feed input that the buffer cannot store, and overwrite areas that hold executable code, replacing it with their own code. For example, an attacker can overwrite a pointer (an object that points to another area in memory) and point it to an exploit payload, to gain control over the program.

Types of Buffer Overflow Attacks

Stack-based buffer overflows are more common, and leverage stack memory that only exists during the execution time of a function.

Heap-based attacks are harder to carry out and involve flooding the memory space allocated for a program beyond memory used for current runtime operations.

PHP Security Best practices

PHP is the most criticized scripting language when it comes to security. A major chunk of developers and QA experts think PHP has no robust techniques to secure applications. The verdict has some ground too because PHP is the oldest and widely used language for web app development. But for a long time since PHP 5.6, we haven't seen any major updates regarding security and hence the language faces some security issues.

Security Issues in PHP CMS

Popular CMS like WordPress, Joomla, Magento, and Drupal are built in PHP and according to Sucuri, most of the vulnerabilities in PHP CMS came to light during the year 2017:

- WordPress security issues rose from 74% in 2016 Q3 to 83% in 2017.
- Joomla security issues have dropped from 17% in 2016 Q3 to 13.1% in 2017.
- Magento security issues rose marginally from 6% in Q3 2016 to 6.5% in 2017.
- Drupal security issues dropped slightly from 2% in Q3 2016 to 1.6% in 2017.

PHP SECURITY BEST PRACTICES

Possible solutions of PHP Security

- ✓Update PHP Regularly
- ✓Cross site scripting (XSS)
- ✓SQL Injection Attacks
- ✓Cross site request forgery XSRF/CSRF
- ✓Session Hijacking
- ✓Hide Files from the Browser
- ✓Securely Upload Files
- ✓Use SSL Certificates For HTTPs
- ✓Deploy PHP Apps on Clouds

MCQ

1. What is the name of the program that converts Java byte code into Dalvik byte code?

A. Android Interpretive Compiler (AIC)

B. Dalvik Converter

C. Dex compiler

D. Mobile Interpretive Compiler (MIC)

2. Definition of Loader?

A. loaders make it easy to asynchronously load data in an activity or fragment

B. loaders make it easy to synchronously load data in an activity or fragment

C. loaders does not make it easy to asynchronously load data in an activity or fragment

D. None of the above

3. Android is based on Linux for the following reason.

A. Security

B. Portability

C. Networking

D. All of these

4. Which among the following are part of "Application" layer of Android Architecture

A. Contacts

B. Browser

C. Phone

D. All of these

5. Which company developed android?

A. Apple

B. Google

C. Android Inc

D. Nokia

REFERENCES

- https://www.imperva.com/learn/application-security/buffer-overflow/.
- https://www.cloudways.com/blog/php-security/