# BUFFER OVERFLOW

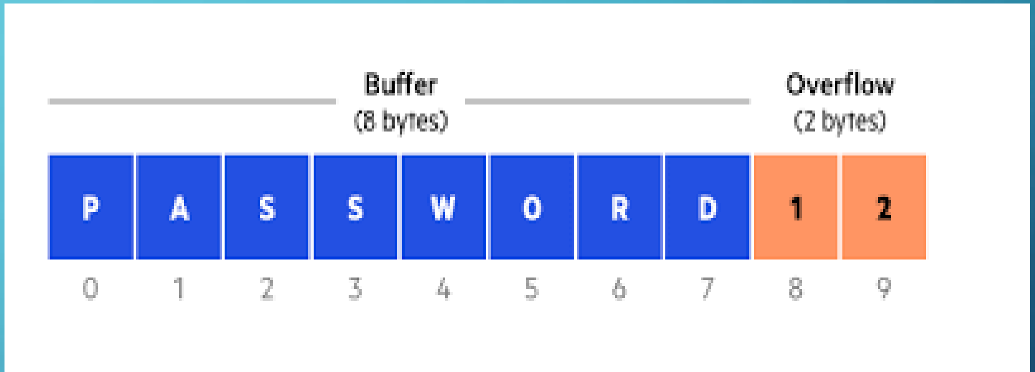
BY WALKER POSTON

# WHAT IS IT?

- Buffers are memory storage places that temporarily hold data while it is being moved from one location to another.

- A buffer overflow is the result whenever the volume of data exceeds the storage capacity of the memory buffer.

- Buffer overflows can affect all types of software and typically are the cause of malformed inputs and failure to allocate enough space for the buffer.



Buffer (8 bytes)

Overflow (2 bytes)

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| P | A | S | S | W | O | R | D | 1 | 2 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

# WHAT TYPE OF VULNERABILITY IS IT?

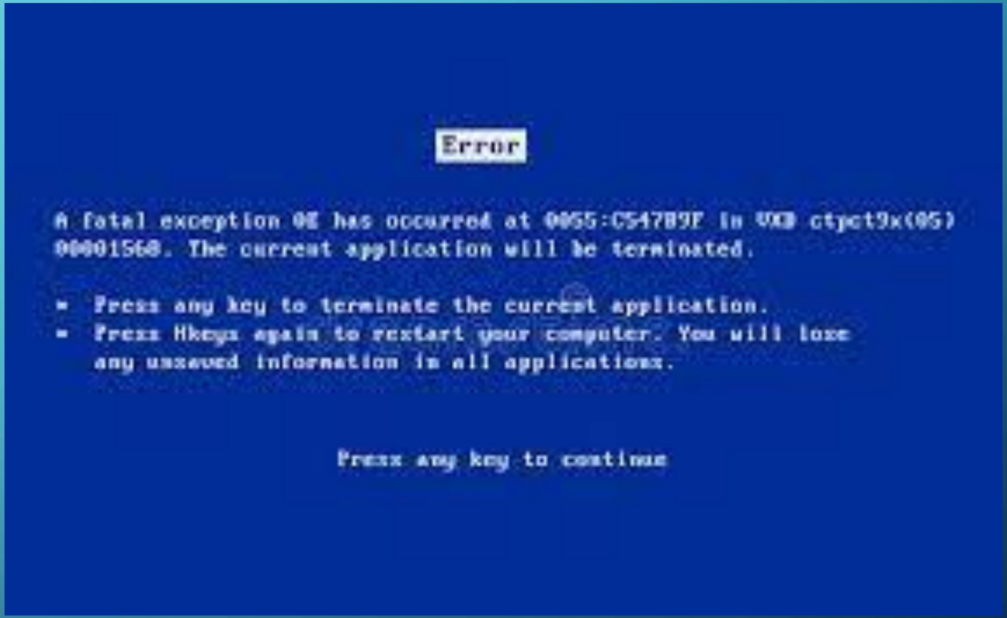
- A buffer overflow is considered to be a software security vulnerability.
- Most software developers are aware of this vulnerability, but this attack is still quite common among legacy and newly developed applications.
- The problem with buffer overflow is there are many ways one can occur and the another problem is the error prone techniques used to prevent them.
- This vulnerability is not an easy one to discover and when one is found, it is pretty difficult to exploit.

# HOW DOES IT WORK?

- In a classic buffer overflow exploit, the attacker sends data to the program, which is stored in an undersized stack buffer.
- The result is that information on the stack is overwritten, including the function’s return pointer.
- The data sets the value of the return pointer so that when the function returns, it transfers control to malicious code contained in the attackers data.
- Although this type of stack buffer overflow is still common on some platforms and in some development communities, there are a variety of other types of buffer overflow, like Heap buffer overflow.

# HOW DOES ARCHITECTURE PLAY INTO THE ISSUE?

- The techniques to exploit a buffer overflow vulnerability vary by architecture, operating system, and memory region.
- A buffer overflow occurs when more data is put into a fixed length buffer than the buffer can handle.
- This extra information has to go somewhere and can overflow into adjacent memory space, corrupting or overwriting the data held in that space.
- This can result in a system crash.



```
                         Error

A fatal exception 0E has occurred at 0055:C54789F in VXD ctpct9x(05)
00001568. The current application will be terminated.

*  Press any key to terminate the current application.
*  Press Hkeys again to restart your computer. You will lose
   any unsaved information in all applications.


                 Press any key to continue
```

# WAYS TO PREVENT BUFFER OVERFLOW

- Developers can protect against buffer overflow vulnerabilities via security measures in their code, or by using a language that offers built-in protection.
- Modern operating systems also have run-time protection and some of the protections include: Address space randomization(ASLR), Data execution prevention, and Structured exception handler overwrite protection (SEHOP).
- Security measures in the code and operating systems help but aren't always the best. Once a buffer vulnerability is discovered, it must be patched immediately and the organization must make sure that all users have access to the patch.

# HOW CONCERNED SHOULD WE BE?

- As long as the correct precautions are taken to prevent a buffer overflow attack from occurring, like the examples mentioned in the previous slide, then there should not be a concern.
- The only time one should be concerned about a buffer overflow attack is when they do not take these precautions or use software that has no protection against this vulnerability and when they find a vulnerability and do not patch it.
- Use modern operating systems and applications proven to protect against buffer overflow and there should be nothing to be concerned about.

# REFERENCES

*Buffer Overflow | OWASP Foundation*. (n.d.). OWASP. https://owasp.org/www-community/vulnerabilities/Buffer_Overflow#:%7E:text=Buffer%20overflow%20is%20probably%20the,applications%20are%20still%20quite%20common.

Klepfish, N., Hasson, E., Lynch, B., McKeever, G., Lynch, B., Hewitt, N., V., D., N., & Lynch, B. (2019, December 29). *What is a Buffer Overflow | Attack Types and Prevention Methods | Imperva*. Learning Center. https://www.imperva.com/learn/application-security/buffer-overflow/

Veracode. (n.d.). *What Is a Buffer Overflow? Learn About Buffer Overrun Vulnerabilities, Exploits & Attacks*. https://www.veracode.com/security/buffer-overflow#:%7E:text=A%20buffer%20overflow%2C%20or%20buffer%20overrun%2C%20occurs%20when%20more%20data,data%20held%20in%20that%20space.