Buffer Overflow

Ian Hagmann, Steven Day, Christopher Hossele, Carlos Cruz Suarez

What is Buffer Overflow?

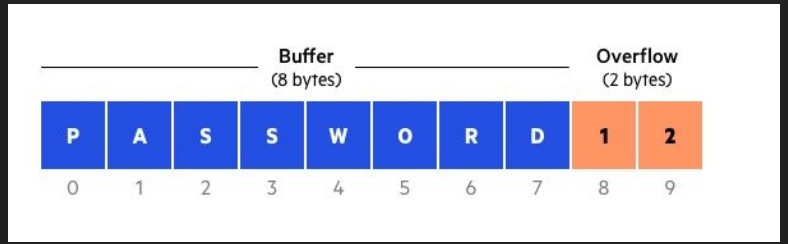
Buffer

- A buffer is a temporary storage option used when transferring data from one place to another.

Buffer Overflow

- A buffer overflow occurs when a specific amount of data is too sizeable for a memory buffer.
- These buffer overflows can occur maliciously if an attacker intentionally inputs a large amount of data that is too large for a specific buffer, or unintentionally if systems are improperly constructed to where large amounts of data are accidentally transferred.

There are several reasons to why these occur, mitigation techniques to prevent, and notable events that are focused on buffer overflows.



How does it happen?

Breakdown:

- Phishing and social engineering: Getting in through human error
- Exploiting Outdated Software: When the code’s old, the hacker will be bold
- Malware Magic: Skimmers and scrapers harvest data in real time

Fun fact: Over 60% of POS breaches come from unpatched software

Notable Events

- Morris Worm- 1988. One of the first major cyber incidents. The attack on exploited buffer overflow allowed it to inject and execute its code on vulnerable systems. It spread uncontrollably.
- The Stagefright attack- 2015. Buffer overflow vulnerability in Android’s media framework. Allowed attackers to execute arbitrary code by sending a specially crafted MMS message. Impacted nearly a billion android devices.
- Yokogawa Centum Buffer Overflow Vulnerability- 2014. Yokogawa distributed control system (DCS) was widely used in critical industrial sectors like energy. Exploited the DCS network to cause crashes or execute codes. Posed risk to operational continuity, safety, and the potential for sabotage.

Mitigation Strategies

- Writing secure code
  - Some functions may be vulnerable to buffer overflow
- Compiler warnings
  - Some compilers warn against insecure functions and suggest replacements
- Stack canaries
  - Random values at run time that are verified after changes to the stack
  - If they are overwritten, the program will stop
- Data execution privilege
  - Can be both hardware and software level
  - Marks code in the stack as non-executable
- Address space layout randomization
  - Randomizes addresses of functions in memory at runtime