Armaan Shamsaasef

Buffer Overflows

Buffer Overflows - SY0-701 - 2.3

Definition: A software vulnerability that occurs when a program writes more data to a fixed-length block of memory (a *buffer*) than it can hold.

How a Buffer Overflow Works

1. The Buffer: A program allocates a fixed amount of memory (e.g., 8 bytes) to store user input, like a username.
2. The Overflow: An attacker sends input that is larger than the allocated buffer (e.g., 16 bytes).
3. The Corruption: The excess data "overflows" the buffer and overwrites adjacent memory.
4. The Consequence: This can corrupt other variables, crash the program, or, most dangerously, overwrite the return pointer that tells the program what to do next.

The Goal: Code Injection

The primary objective of a malicious buffer overflow is to overwrite the return address in the stack's memory. The attacker replaces the legitimate return address with a pointer to their own malicious code (shellcode) that they sent as part of the input.

When the function finishes, instead of returning to the main program, it jumps to and executes the attacker's code.

Key Terms

- Buffer: A temporary, fixed-size area of memory for storing data.
- Shellcode: A small piece of malicious code, often designed to provide a command shell to the attacker.
- Return Pointer: A memory address in the stack that tells the program where to go after a function finishes executing. This is the primary target.

Real-World Example from the Video

- A programmer creates a simple application that asks for a 4-character name.
- The program allocates a 4-byte buffer in memory to store this input.
- An attacker enters the name "ABCDEFGHIJ\x68\x63\x61\x6c\x63".
    - `ABCDEFGHIJ` overflows the buffer and overwrites the return pointer.
    - The hex values `\x68\x63\x61\x6c\x63` are x86 assembly instructions that translate to the Windows command `calc` (launch the calculator).
- By precisely overflowing the buffer, the attacker overwrites the return address to point to their "calc" shellcode.
- When the function ends, instead of exiting normally, the program executes the shellcode and launches the calculator.

This demonstrates that if an attacker can launch `calc`, they can also launch `cmd.exe` to get a full command prompt and take control of the system.

Summary

- A Buffer Overflow is an anomaly where a program overruns a buffer's boundary and writes to adjacent memory.
- The main security risk is that it can allow an attacker to inject and execute arbitrary code on the target system.
- The exploit works by overwriting the return pointer on the stack to redirect program flow to the attacker's malicious shellcode.
- This is a classic vulnerability that modern defenses like Data Execution Prevention (DEP) and Address Space Layout Randomization (ASLR) are designed to mitigate.

Buffer overflow

Variable Name	A								B	
Value	'e'	'x'	'c'	'e'	's'	's'	'i'	'v'	25856	
Hex Value	65	78	63	65	73	73	69	76	65	00