Quinten Lambert

John Hellrung

Buffer Overflow

1. Describe the buffer overflow problem.
   1. Buffer overflow is where a program attempts to write data to a buffer and overwrites outside of the buffers bounds. This then in turn overwrites adjacent, which becomes a case of violation of memory safety.
2. What happens if you exceed the size of an array in Python? Do you consider this robust behavior?
   1. When you exceed the size of an array in python there is an error and the program or service crashes unable to continue its processes. This is a robust behavior because when you exceed the maximum that the server or program can handle then the program or server should cease to serve its contents.
3. List three ways you could potentially overflow a buffer or exceed the size of an array in your program.
   1. Char buf[size\_of\_buff], get(buf). There is no safety for this and the user is dependent on entering a value fewer than the size\_of\_buff characters.
   2. The attacker could overwrite a local variable that is near the buffer in memory on a stack to change the behavior of the program.
   3. Use an integer overload and have it keep iterating through as it would never stop running.
4. How could you prevent a buffer overflow or out of bounds error from occurring in your program?
   1. Put in checks to make sure that the values entered into the program are actually what they should be instead of possibilities to breaking the program as it is.