* There are mainly three ways to find a buffer overflow, those are – fuzzing, reverse engineering, source code review.
* There are many different categories of fuzzing tools and techniques that we can employ based on our particular needs.
* A fuzzer is considered generation-based if it creates malformed application inputs from scratch, following things like file format or network protocol specifications.
* A mutationbased fuzzer changes existing inputs by using techniques like bit-flipping to create a malformed variant of the original input.
* Generally speaking, a fuzzer that is aware of the application input format can be classified as a smart fuzzer.
* A Word About DEP, ASLR, and CFG. Microsoft implements several such protections, specifically Data Execution Prevention (DEP), Address Space Layout Randomization (ASLR), and Control Flow Guard (CFG).
* DEP is a set of hardware and software technologies that perform additional checks on memory to help prevent malicious code from running on a system.
* ASLR randomizes the base addresses of loaded applications and DLLs every time the operating system is booted. On older Windows operating systems like Windows XP where ASLR is not implemented, all DLLs are loaded at the same memory address every time, making exploitation much simpler. When coupled with DEP, ASLR provides a very strong mitigation against exploitation.
* Finally, CFG, Microsoft’s implementation of control-flow integrity, performs validation of indirect code branching, preventing overwrites of function pointers.