Remote code execution has been a vulnerability with serious consequences in software-based systems and applications and it occurred from time to time. Remote code execution can be a result of buffer overflow attack, when data supplied remotely are written to a memory area to be executed. In general, buffer overflow attack attempts to achieve two goals: 1) injecting the attack code through hardcoded inputs in programs, inputs from command line or network input redirection via sockets; 2) changing the execution path of the running process by overwriting the return address with the address that points back to the buffer where the malicious codes have been injected so that the injected code will be executed (shown in Fig. 4a) [26][27]. The code may self-propagate or give attacker control over the machine. Attacker may set stack pointer to return to a dangerous library functions, such as system(), exec(), or to a malicious code inserted by the attacker (Fig. 4b). Buffer overflow countermeasures, such as write secure code (e.g. avoid strcpy function), stack execute invalidation (e.g. Linux kernel enforces the nonexecutable stack), compiler tools (e.g. StackShield copies the return address to a safer place; StackGuard detects and defeats smash stacking attacks by protecting the return address on the stack from being altered), dynamic run-time check (libsafe library provides a way to secure calls to these functions and ensures that address is not overwritten), etc. have been explained in [28]. Students will learn what buffer overflow is, how it can be exploited to achieve remote code execution, and what countermeasures can be taken to avoid it. Student will be demonstrated a proof of concept attack using the EternalBlue exploit, which targets a remote code execution vulnerability CVE-2017-0144 in Microsoft SMBv1 server, which is mainly used to provide file and printer sharing on Microsoft Windows Networks. This vulnerability led to the spread of several cyberattacks, including the computer worm EternalRocks and ransomware WannaCry. Students will be provided with source code for the proof of concept attack above and conduct hands-on experiment with the attack. Students will then be asked to build their own payload (the code to be executed on the remote machine) to design a new attack using this vulnerability.