Stack Guard:

Stack guard is technique of automatic detection and prevention of Buffer overflow attacks. It is a compiler extension that enhances the executable code produced by the compiler so that it detects and thwarts the buffer-overflow attacks against the stack.

Stack guard enhanced programs detects the behavior of writing to the return address of a function while it is still active. Changes to active return addresses are blocked by StackGuard either by detecting the update before the feature returns or fully blocking the write to the return address. Detecting changes to the return address is a more effective and portable method, while stopping the change is more secure. Both methods are supported by StackGuard, as well as adaptive switching between them.

So basically the model of opration of Stack Guard could br formalized as, whenever the buffer overflow is detected (by any of the above two methods), the process is terminated. Since an unknown amount of state has already been compromised before the attack is detected, it is difficult to safely restore the process's state. So the process must exit. As a consequence, the method exits only with static data and code, eliminating any potential attacker corruption.

ASLR Protection:

Address Space Layout Randomization (ASLR) is a memory-protection process for operating systems that protects against buffer-overflow attacks. It assists in ensuring the memory addresses associated with operating processes on systems are unpredictable, making bugs or weaknesses in these processes more difficult to exploit.

Through randomizing the offsets used in memory layouts, ASLR strengthens a system's control-flow integrity by making it more difficult for an attacker to conduct a successful buffer-overflow attack. ASLR does much better on 64-bit systems, which have significantly more entropy (randomization potential), so brute force attacks are much difficult to make success whereas in 32-bit systems it is relatively easier to throw attack in a brute force manner.

Some limitations of ASLR includes that it effectively does not resolve vulnerabilites, it just helps in making exploits much more difficult and challenging. Moreover, it also does not track or report vulnerabilities and doesn’t offer any protection for binaries which are not compiled and built with the ASLR support.