Deniz Aytemiz

Paper Summaries

1. Just-In-Time Code Reuse: On the Effectiveness of Fine-Grained Address Space Layout Randomization

   Address space layout randomization is a defense against code reuse attacks. In this paper writers state that fine grained ASLR may not be significantly more effective than traditional ASLR which can be bypassed by attackers. Writers show evidence of this by implementing just in time code strategy in which they can adapt a multiple times use random memory and map a vulnerable application memory layout to make just in code use attack possible.

2. Control-Flow Integrity

   In this paper a defense measure called control-flow integrity for attacks targeting machine code subversion is discussed. It is stated that runtime control flow would prevent many software exploits today. It aims to be embedded in software both control-flow policy enforced in run time and enforcement mechanism. Paper demonstrates how CFI is enforced and implemented on Windows x86 by supporting experimental data. It is shown that CFI is effective, simple and compatible with many existing software in addition to easily implemented on modern processors.