Monitor for DMA of the Minix Network Driver

Oliver Wilén, Marcus Dypbukt Källman, Marco Godow, Sebastian Veijalainen

Introduction

The network driver that Qemu uses for Minix has a feature that gives the network driver Direct Memory Access (DMA). DMA is used to bypass the CPU and access system memory directly. When a CPU is using traditional I/O when a read or write operation is issued, the CPU is fully occupied during the operation. DMA counters this issue by freeing the CPU to do other tasks while the transfer is in process. This gives DMA an obvious performance advantage in some situations. However, by granting direct access to the system memory, DMA opens security vulnerabilities. Our goal with this project is to improve the security of DMA by using a monitor, which provides additional constraints on the network driver.

Targeted Vulnerabilities

In DMA attacks a compromised device (or external device) can gain DMA and be able to read and write directly to memory without the supervision of the operating system. A compromised peripheral device could read or write parts of memory it should not access when running as it normally should. Not allowing physical access to exposed ports can be used to prevent DMA attacks. DMA can also be disabled for ports that are not in use or when the device is locked.

Minimal

1. Monitor memory access of Minix network driver. (Specifically the dp8390 used by Qemu
2. Be able to stop the driver from accessing the memory(No certain conditions needed). E.g. completely turn off DMA.

Optional

1. Detect wrongful memory access behaviour by the driver according to its specification.
2. Change some parts of the driver code without affecting our monitor's functionality.
3. Develop a DMA attack to test our solution.