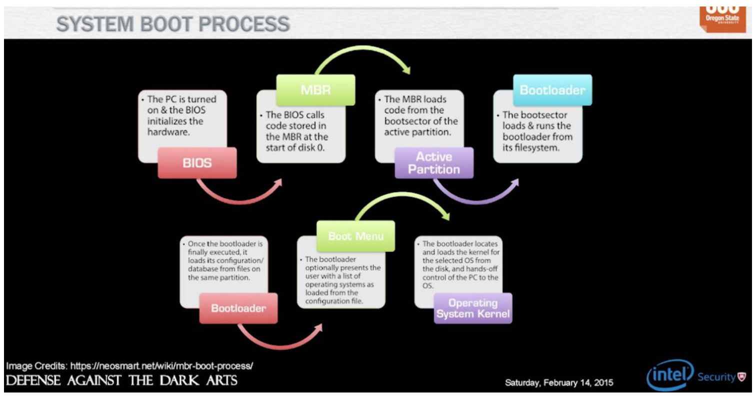
Nicolas Barraclough
10/30/2020
CS373 - Defense Against the Dark Arts

Week 5 Write-Up

In week 5, we learned about rootkits and bootkits, kernel security, and how these topics relate to malware and the Windows Operating system. I learned a little bit about rootkits in Intro to Security and a little about threads and processes in Operating Systems, but most of this week's material was brand new to me.

Rootkits are mostly prevalent in Windows 32-bit systems, but they exist in 64-bit systems as well. This form of malware tries to compromise the system at the kernel-mode level so that it can conceal its existence and actions from users and system processes so that they can have full, undetected control over the system. This stealth allows the malware to exist on a system for long periods of time. They do this by doing what's called "hooking" where the malware intercepts a valid instruction of the system and redirects to its own code. For example, when the system wants to check the amount of processes running on the system, a rootkit might hook onto that call and redirect to its own code that will hide its existence. By hooking the system calls that would otherwise reveal the malware to the system or the user, the malware can trick the system into thinking it's not there. As of 2015 when this lecture was filmed, rootkits were used by approximately 10% of existing malware. Aside from any malicious intent, rootkits can also provide one of the best ways to learn about kernel security challenges.

Bootkits are pretty similar to rootkits in that they work to have full access/control to the system, however, they do so in a different way. Bootkits will hook system calls by infecting the master boot record. The system will think that the pre-infected boot record was run when in fact, the malware infected the boot record and hooked system calls to stay undetected.



SYSTEM BOOT PROCESS

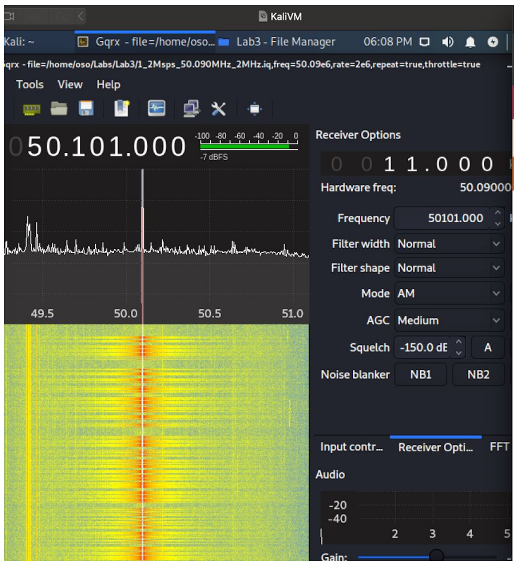
- The PC is turned on & the BIOS initializes the hardware.
- The BIOS calls code stored in the MBR at the start of disk 0.
- The MBR loads code from the bootsector of the active partition.
- The bootsector loads & runs the bootloader from its filesystem.
- Once the bootloader is finally executed, it loads its configuration/ database from files on the same partition.
- The bootloader optionally presents the user with a list of operating systems as loaded from the configuration file.
- The bootloader locates and loads the kernel for the selected OS from the disk, and hands-off control of the PC to the OS.

Image Credits: https://neosmart.net/wiki/mbr-boot-process/
DEFENSE AGAINST THE DARK ARTS
Saturday, February 14, 2015

Recovering from a rootkit is not as easy as removing the malware-associated files from the system. Rootkits and bootkits can overwrite system information and system files so that if the system reboots or the files are not properly restored, the malware can reestablish itself and resume its malicious intent - once again concealing its existence.

After learning about rootkits and the ways in which they hide themselves, we learned how to manually patch a rootkit by isolating the malware in memory and cutting off any pointers pointing to the malware. So long as there is not a thread that will overwrite the patch and create another pointer to the malware, the malware will still exist on the system, but it will not be reachable and its code will never be seen by the system.

I was going to talk more about the tools that I learned about this week, but with all of the time it took to get everything installed, I have barely enough to talk generally about them. One of the coolest tools I've learned in this class so far is Gqrx. In the lab, we used the heat signature map in Gqrx to translate morse code.


```
gqrx - file=/home/oso/Labs/Lab3/1_2Msps_50.090MHz_2MHz.iq,freq=50.09e6,rate=2e6,repeat=true,throttle=true
```
050.101.000

Receiver Options
0 011.000
Hardware freq: 50.09000
Frequency: 50101.000
Filter width: Normal
Filter shape: Normal
Mode: AM
AGC: Medium
Squelch: -150.0 dE

I'm curious what else can be done with this tool and I'm definitely going to try again at sample 3 when I have the time. This week was the most informative and fun week yet in this course!