

## Memory Forensics of Windows Kernel and User mode Rootkits(WIP)

Carson Harmon and Prof. Marcus Thompson

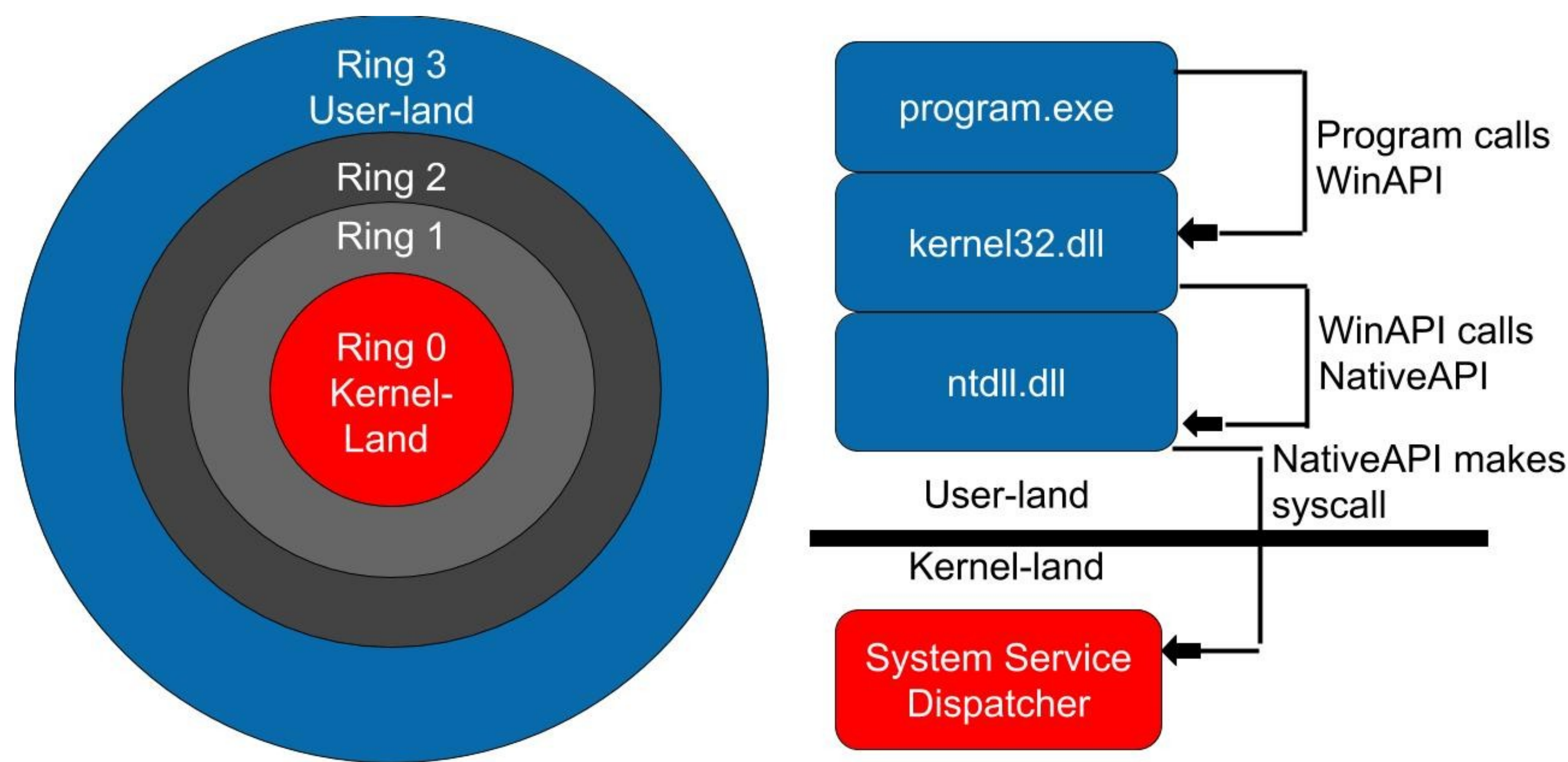
### Abstract

Malware is becoming more sophisticated every year. As forensic and security professionals pioneer new heuristics for discovering malware, malware authors find new ways to remain hidden. Rootkits are now included as modules in other malware to prevent detection and removal. Rootkits operate in either kernel-mode or user-mode. Rootkits operating in kernel space have more control over the infected system, but operating in user space allows the rootkit to function without accessing the kernel directly. The purpose of this research is to identify signatures the two types of rootkit create in memory.

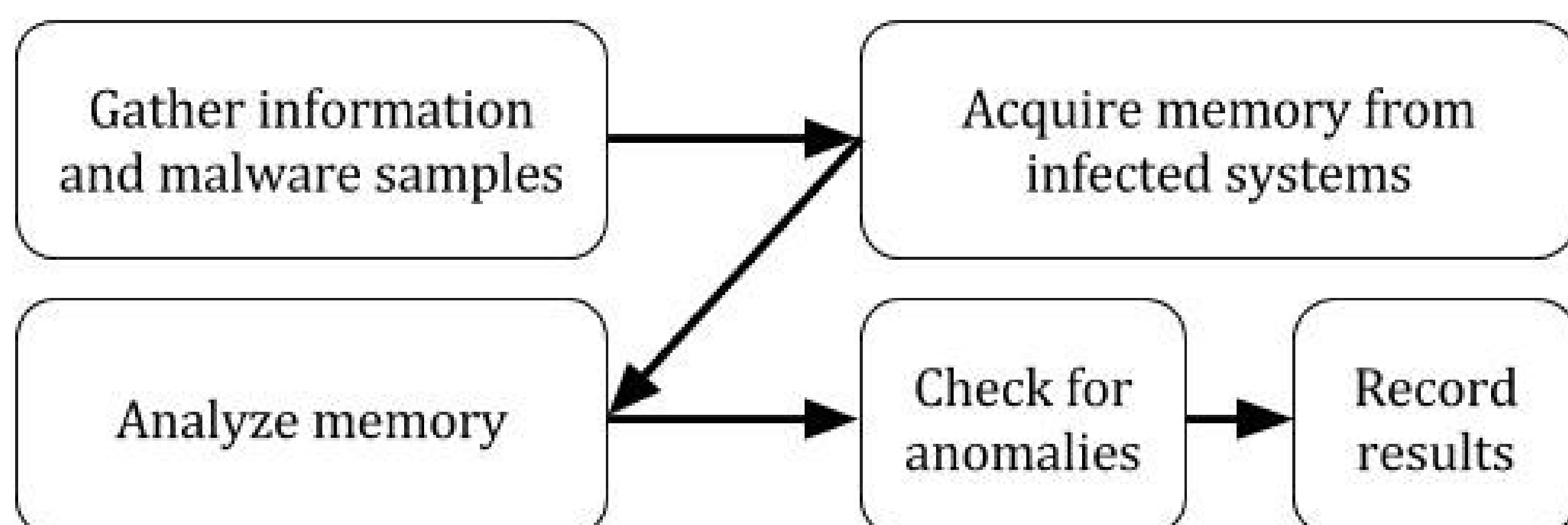
### Research Question

What are the differences in memory between kernel-mode and user-mode rootkits?

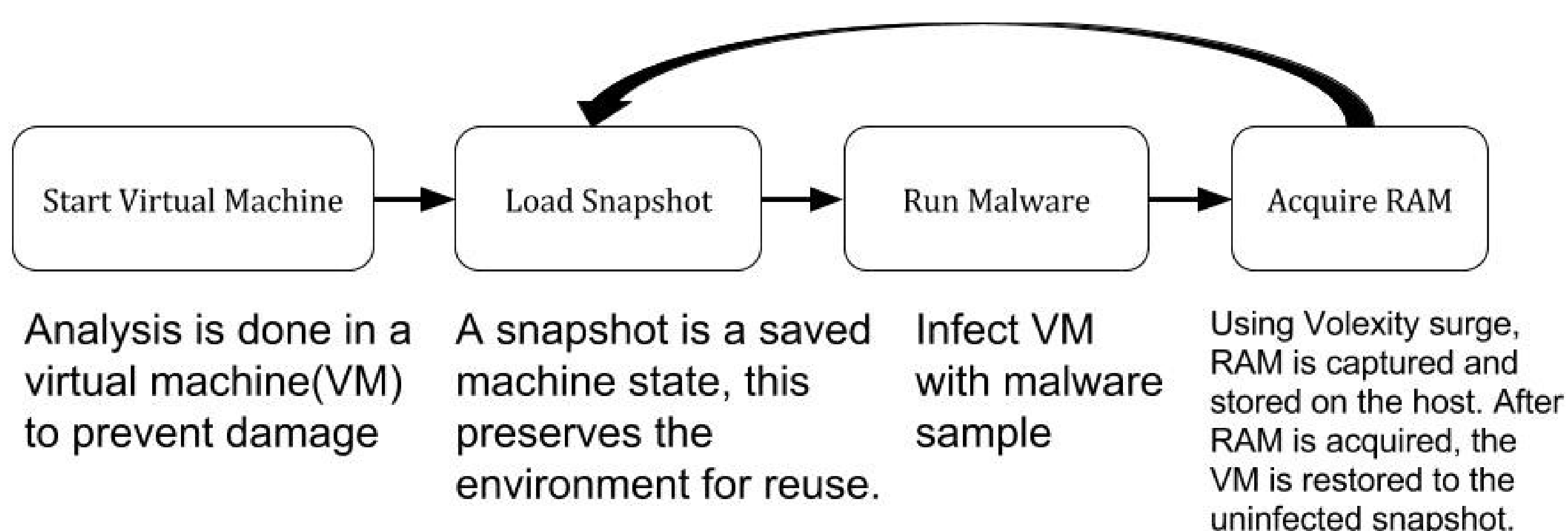
### Windows Process Privilege Levels



### Methodology



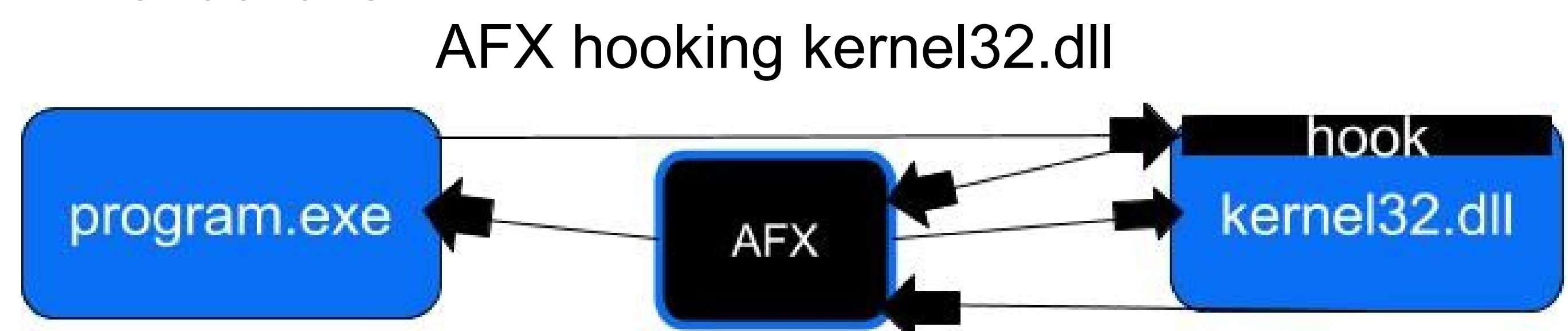
### Memory Acquisition Process



### Example Kernel-mode and User-mode Stealth Techniques

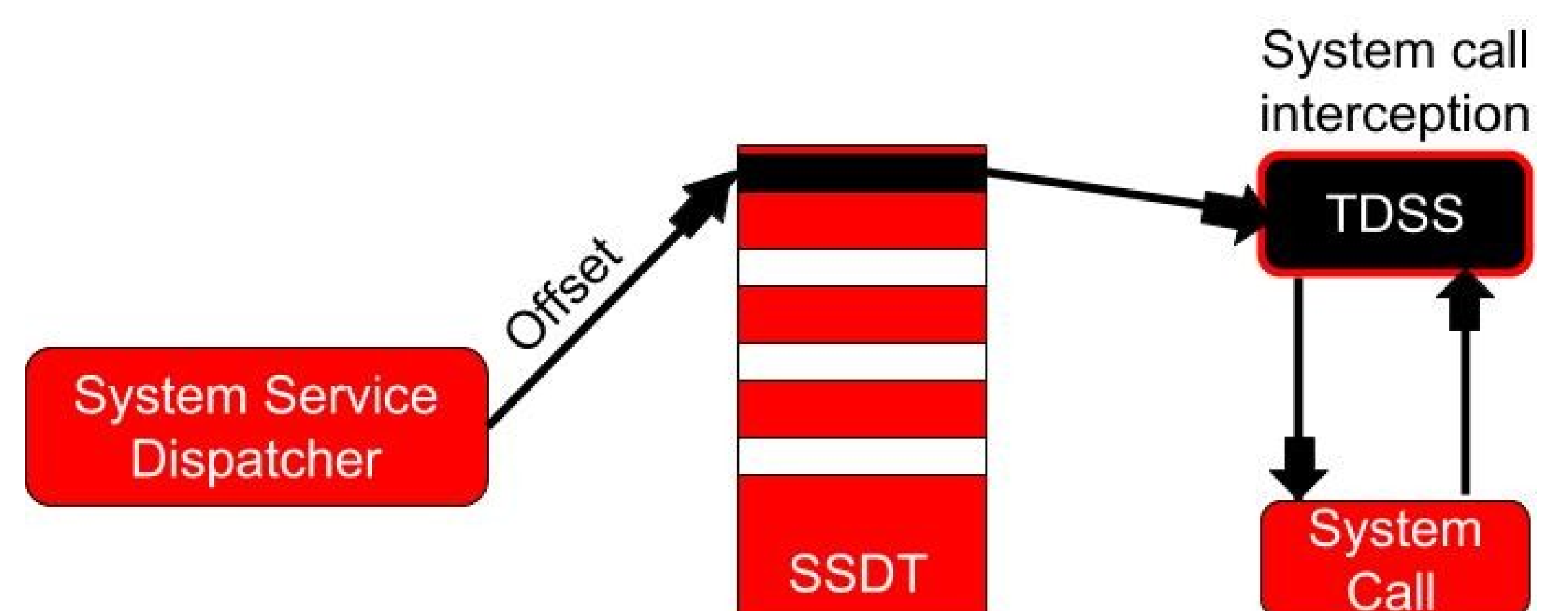
#### AFX

- User-mode rootkit
- Hides registry keys, files, processes
- Performs in-line hook on WinAPI
- Attacks the address space of a process and inserts a JMP instruction in the first few instructions.



#### TDSS (Alureon)

- Kernel-mode rootkit
  - Hides registry keys, files, processes
  - Intercepts System Service Dispatch Table (SSDT)
  - Directly modifies the kernel data structure
- TDSS SSDT interception



### Example Comparison

Both AFX and TDSS hide registry keys, files, and processes. Using volatility, it is possible to enumerate objects in RAM. Although they accomplish the same task, kernel rootkits leave a big signature in memory forensics by directly modifying the kernel. AFX leaves a very small signature by modifying a handful of assembly instructions. Final results will be the result of several kernel-mode and user-mode rootkits. Other samples include HackerDefender and Stuxnet.