

Rootkits – Advanced Malware

Presented by Darren Bilby
Brightstar, IT Security Summit, April 2006

Disclaimer

- This presentation is not designed to scare you (but it might)
- Using these tools within your organisation without explicit permission is a bad idea



Overview

- Introduction
- What is a rootkit?
- How rootkits work
- Rootkit capabilities
- Rootkit demo
- Detection methodologies
- Detection demo
- Mitigations
- Hardware rootkits
- Conclusion



Sony Rootkit

- **First mainstream media coverage of a rootkit**
- **Discovered by Mark Russinovich when using his rootkit detection software**
- **Sony used “rootkit” technology to protect their copy protection mechanism from users**
 - Anything that was named \$SYS was hidden from the system, even the Administrator



What is a Rootkit?

- **“A rootkit is a tool that is designed to hide itself and other processes, data, and/or activity on a system.” – G. Hoglund (www.rootkit.com)**
- **A toolkit used for preservation of remote access or “root”**
- **“A tool used to protect backdoors and other tools from detection by administrators”**
- **A rootkit is not**
 - An exploit of any kind
 - A virus or worm



Rootkits - Why Should You Care?

- **Your current methods for investigating a suspicious machine could be defunct**
- **If you can't detect a backdoor on any given machine, how do you know your machine is clean?**
- **New viruses will use new rootkit technology**



Rootkits - How They Work

- To hide in a system you have to control a system
- Act as a gatekeeper between what a user sees and what the system sees
- Whoever hooks lowest wins
- Requires administrator privileges to install



Rootkits – How They Work

- **To hide what is taking place an attacker wants to:**
 - Survive system restart
 - Hide processes
 - Hide services
 - Hide listening TCP/UDP ports
 - Hide kernel modules
 - Hide drivers



Levels of Access in Windows

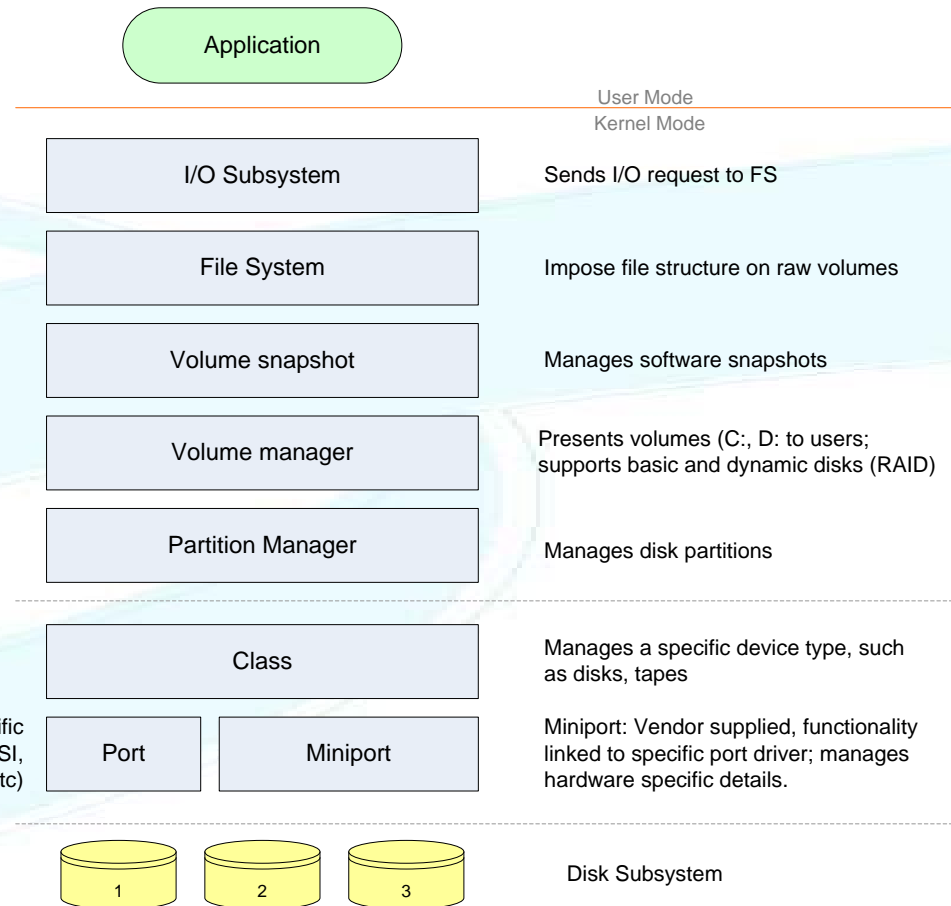
- **Ring 3 – User Land**

- User
- Administrator
- System

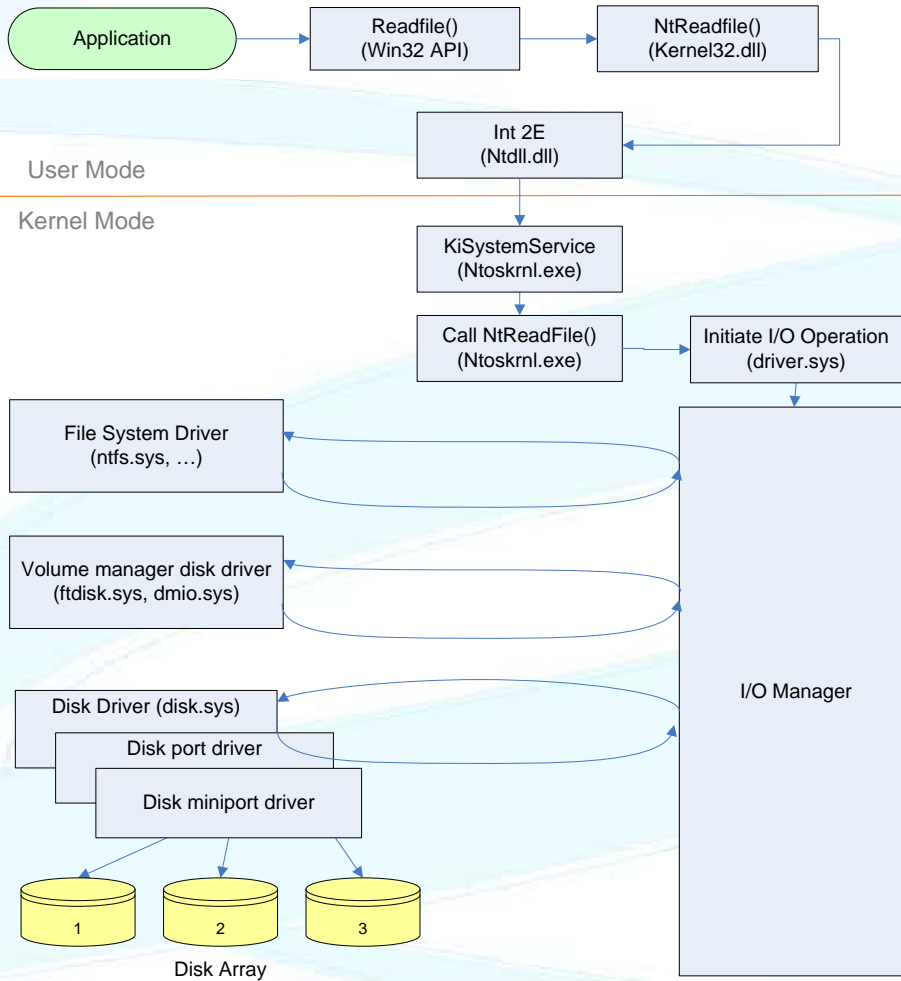
- **Ring 0 – Kernel Land**

- Drivers

Port: Manages a specific transport (SCSIport for SCSI, Storport for RAID and FC, etc)



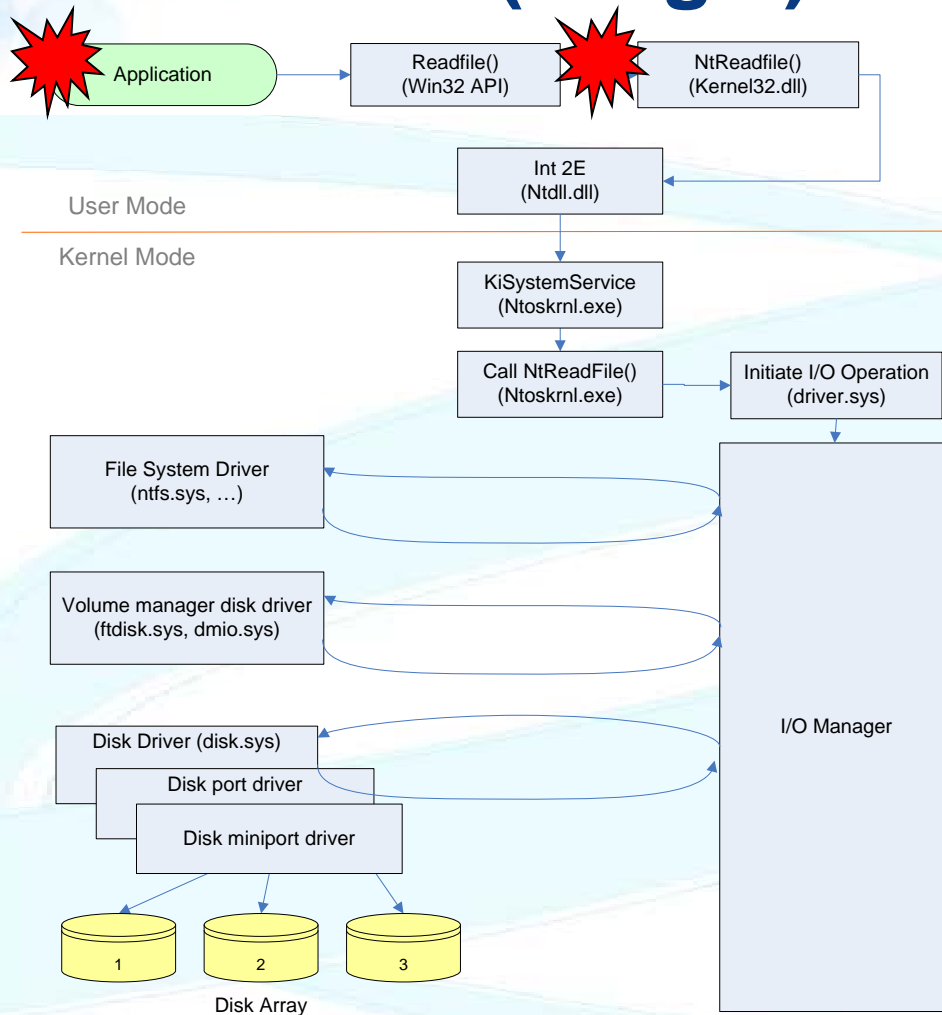
What Happens When You Read a File?



- Readfile() called on File1.txt
- Transition to Ring 0
- NtReadFile() processed
- I/O Subsystem called
- IRP generated
- Data at File1.txt requested from ntfs.sys
- Data on D: requested from dmio.sys
- Data on disk 2 requested from disk.sys

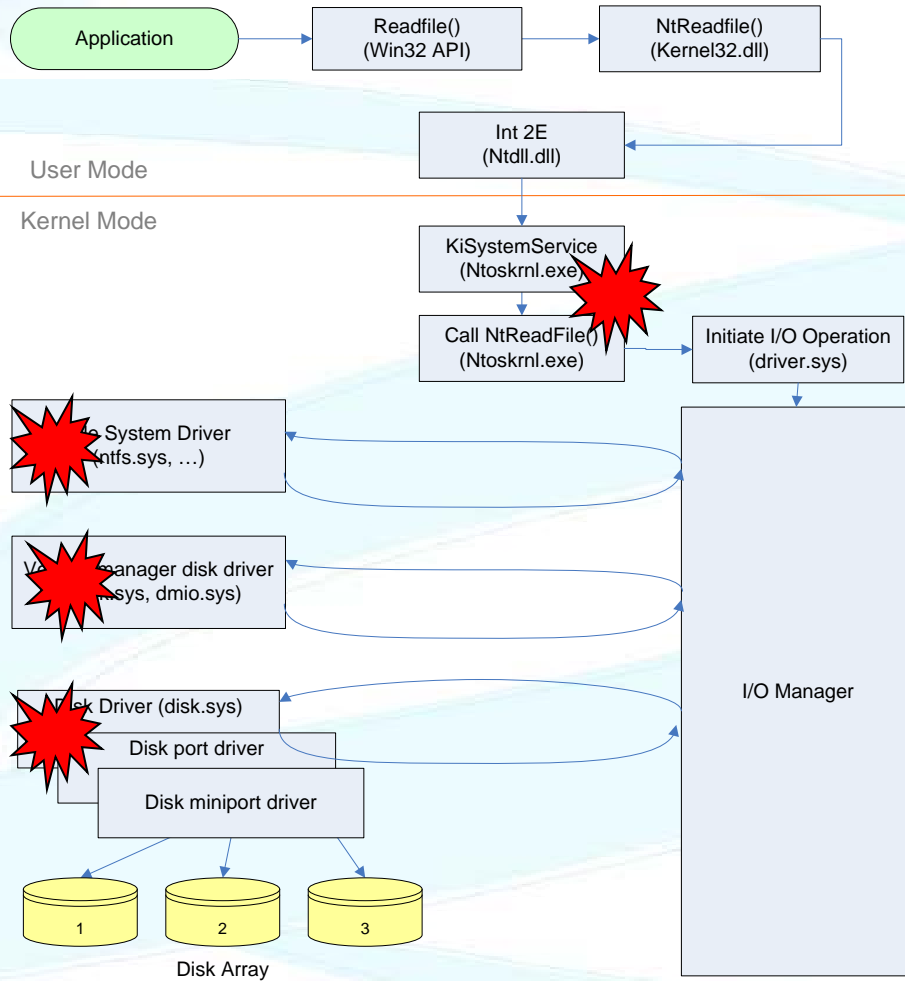


Userland (Ring 3) Rootkits



- Binary replacement eg modified Exe or DLL
- Binary modification in memory eg He4Hook
- User land hooking eg Hacker Defender
 - IAT hooking

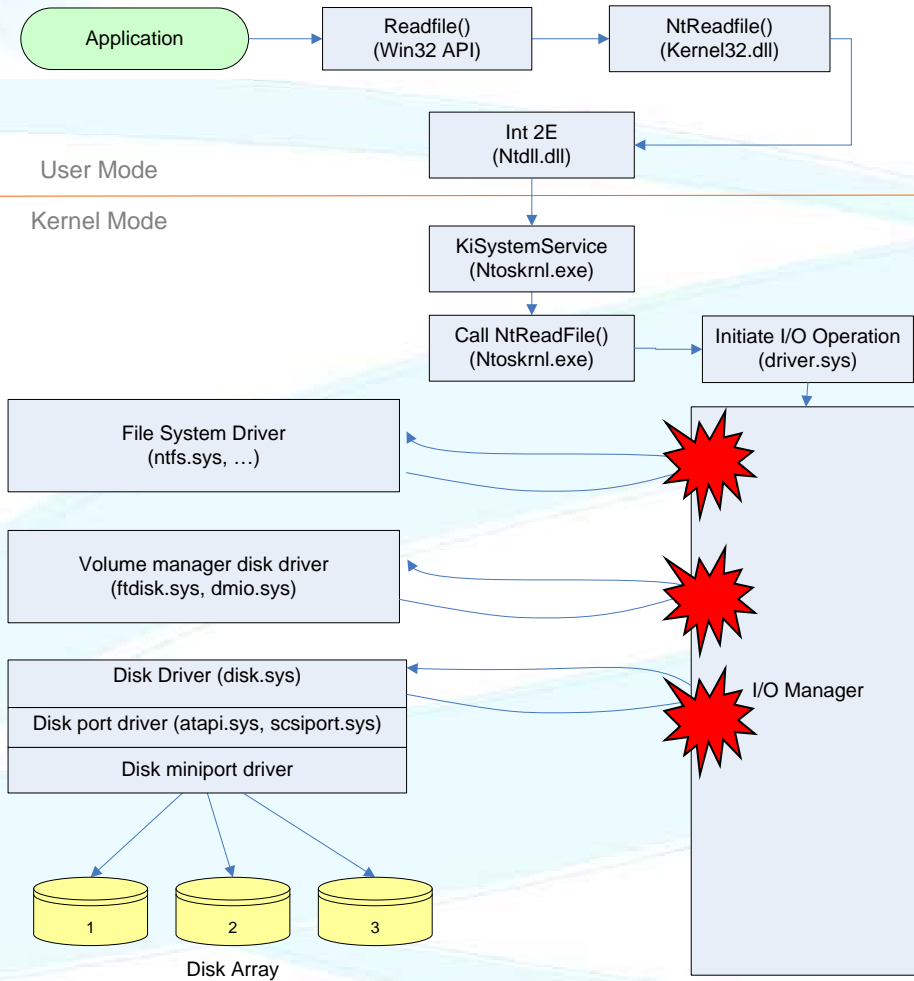
Kernel (Ring 0) Rootkits



- **Kernel Hooking**
E.g. NtRootkit
- **Driver replacement**
E.g. replace ntfs.sys with ntfs.sys
- **Direct Kernel Object Manipulation – DKOM**
E.g. Fu, FuTo



Kernel (Ring 0) Rootkits

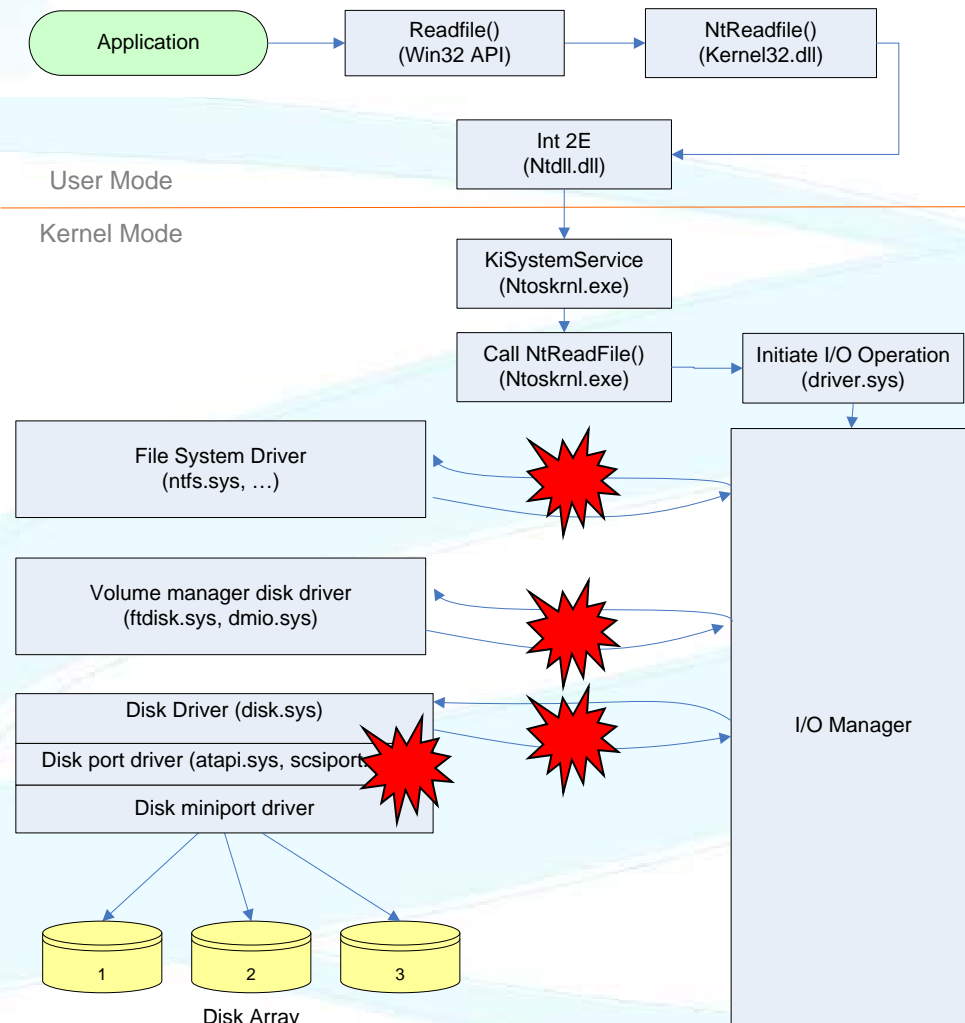


- **IO Request Packet (IRP) Hooking**
 - IRP Dispatch Table

E.g. He4Hook (some versions)



Kernel (Ring 0) Rootkits



- **Filter Drivers**
 - The official Microsoft method
- **Types**
 - File system filter
 - Volume filter
 - Disk Filter
 - Bus Filter

E.g. Clandestine File System Driver (CFSD)



Current Rootkit Capabilities

- **Hide processes**
- **Hide files**
- **Hide registry entries**
- **Hide services**
- **Completely bypass personal firewalls**
- **Undetectable by anti virus**
- **Remotely undetectable**
- **Covert channels - undetectable on the network**
- **Defeat cryptographic hash checking**
- **Install silently**
- **All capabilities ever used by viruses or worms**



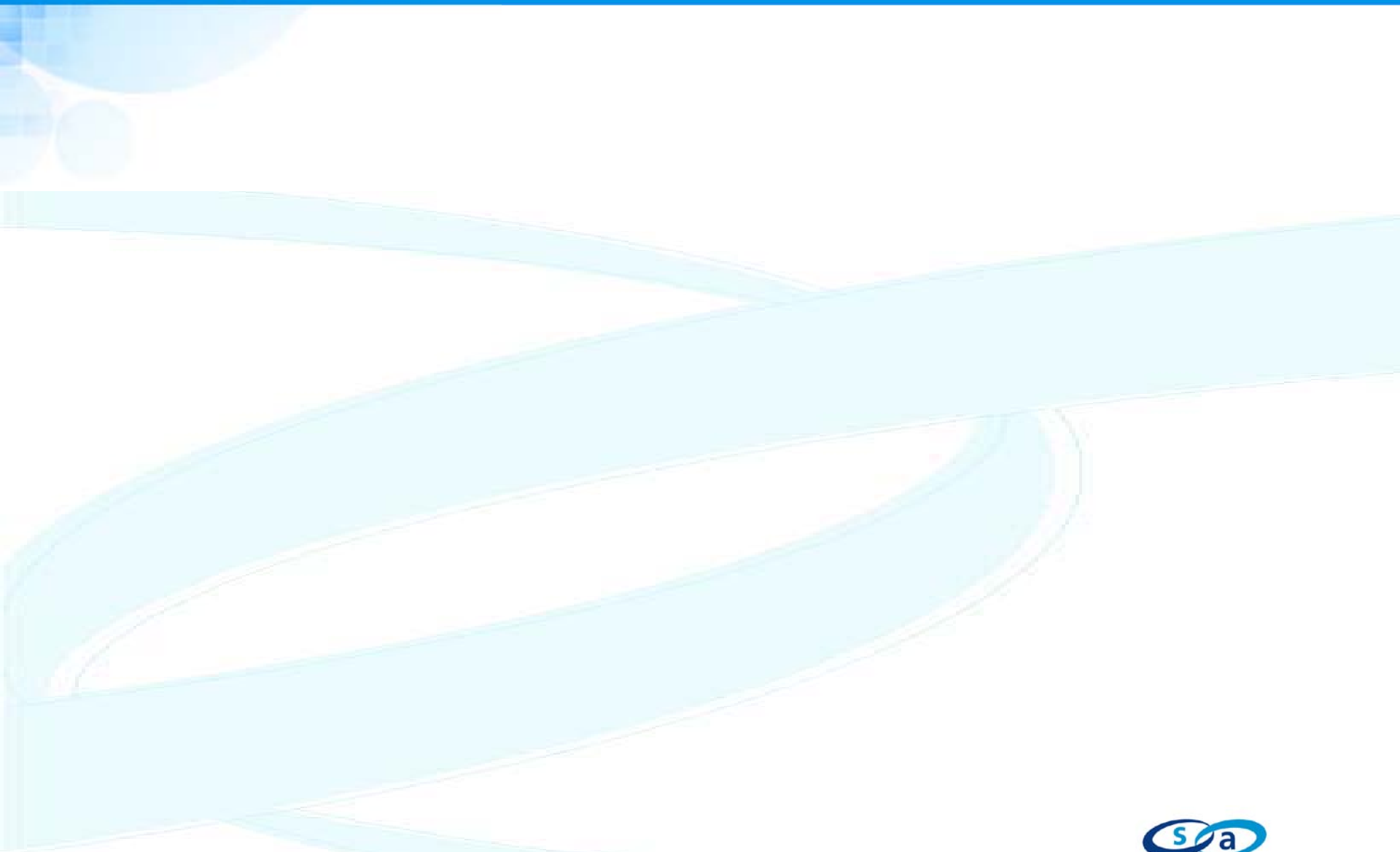
Ring 3 Rootkit: Hacker Defender

- **Hacker Defender**
 - Most widely used rootkit on Windows
 - Hides processes
 - Hides TCP / UDP port bindings
 - Uses simple INI file configuration
 - Easy to detect and remove with defaults
 - Not too difficult to modify to avoid detection
- **Commercial Hacker Defender**
 - No longer available



Hacker Defender - Demo





Ring 3 Rootkit: FU

- **FU Rootkit**
 - Utilises Direct Kernel Object Manipulation (DKOM)
 - Hide processes specified in a file
 - Escalate privileges of processes
 - Hooks calls and rewrites dlls in memory
 - Event Viewer modification
 - Hides device drivers
- **Other examples: He4Hook, Klog, ShadowWalker, Adore, Suckit**



FU Rootkit - Demo



Detecting Rootkits

Common Giveaways

- **Something weird is happening...**
- **Rootkits often cause system instability**
 - Bluescreens on normally stable systems
 - Errors when you attempt to shutdown or reboot
- **Detected bad network traffic**
- **Antivirus/IDS alerts**



Detection Methodologies

- **Traditional Detection**
 - Check integrity of important OS elements against a hash database (sigcheck)
 - Look for unidentified processes (task manager)
 - Check for open ports (netstat)

- **Problems**
 - Can be subverted easily



Detection Methodologies

- **Signature based**
 - Look for known rootkits, viruses, backdoors
 - Antivirus
 - Look for “bad things” living in memory
- **Problems**
 - Requires updated databases
 - Doesn't detect anything it hasn't seen before



Detection Methodologies

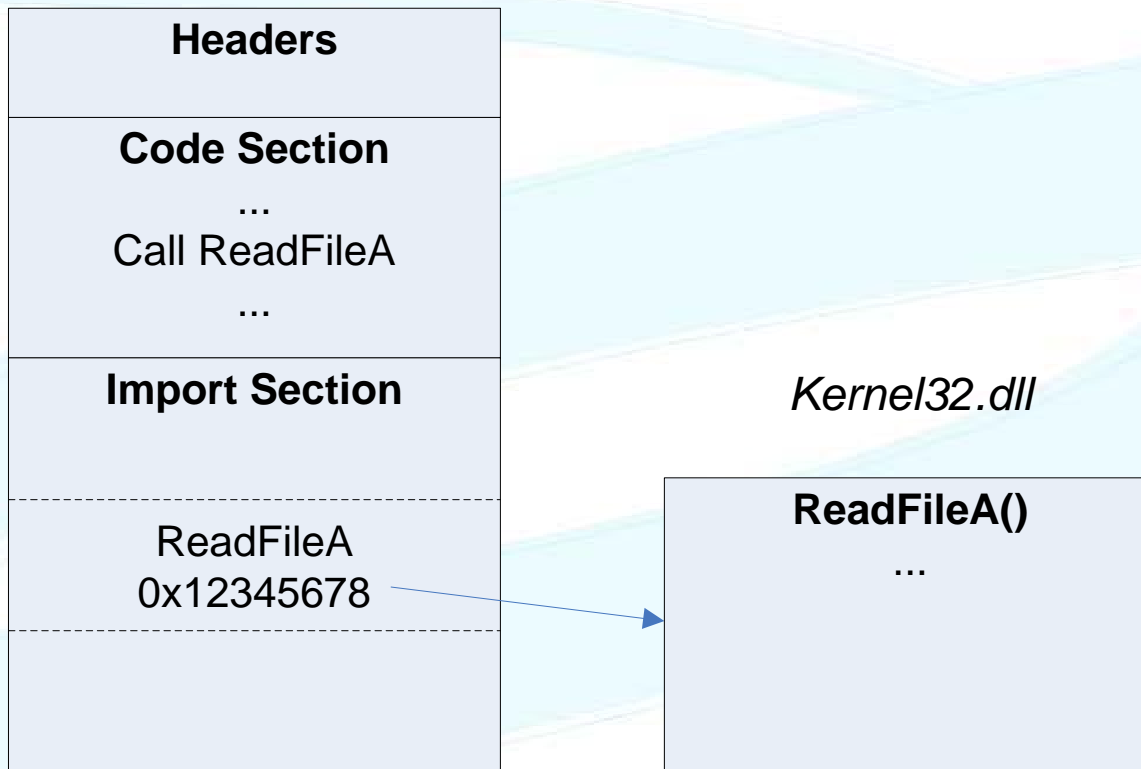
- **Hook detection**
 - Look for modified IAT tables
 - Look for inline hooks
 - Look for modification to important tables
- **E.g.**
 - VICE
 - System Virginty Verifier
 - SDT Restore
 - IceSword
- **Problems**
 - False positives – AV products



How Rootkits Work - Hooking

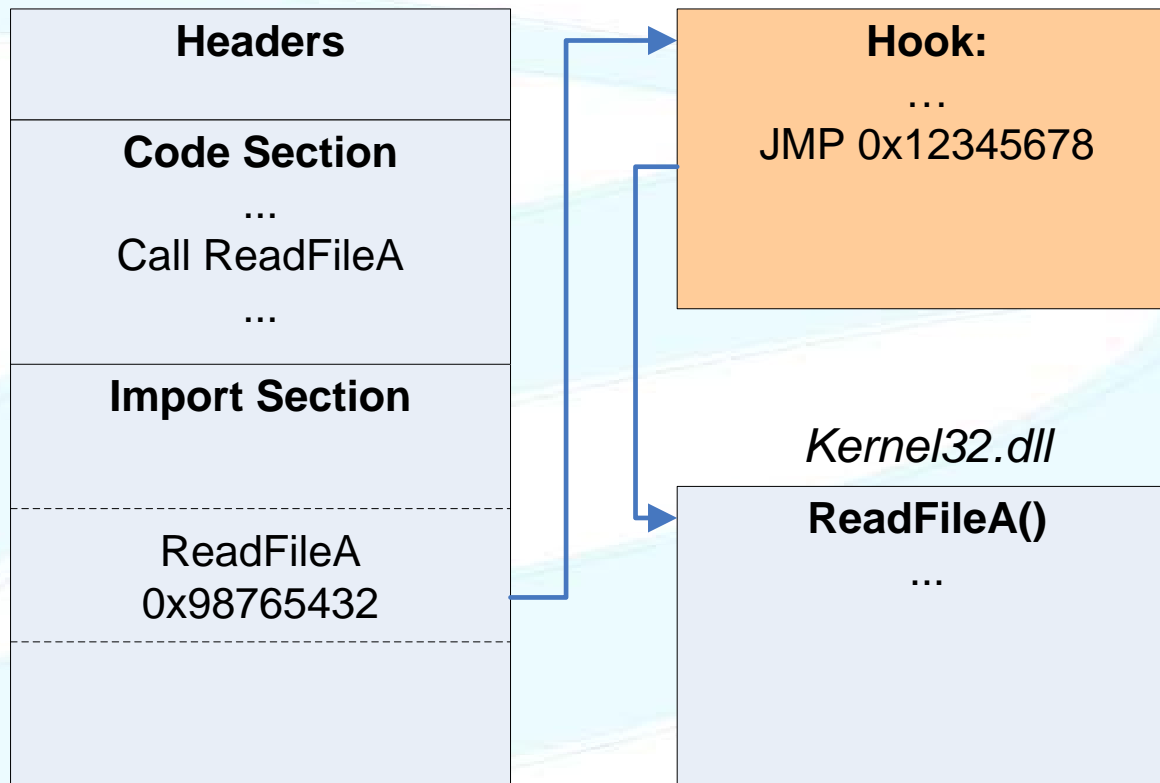
MyApplication.exe

- A standard application



How Rootkits Work - Hooking

MyApplication.exe



- A hooked application



Detection Methodologies

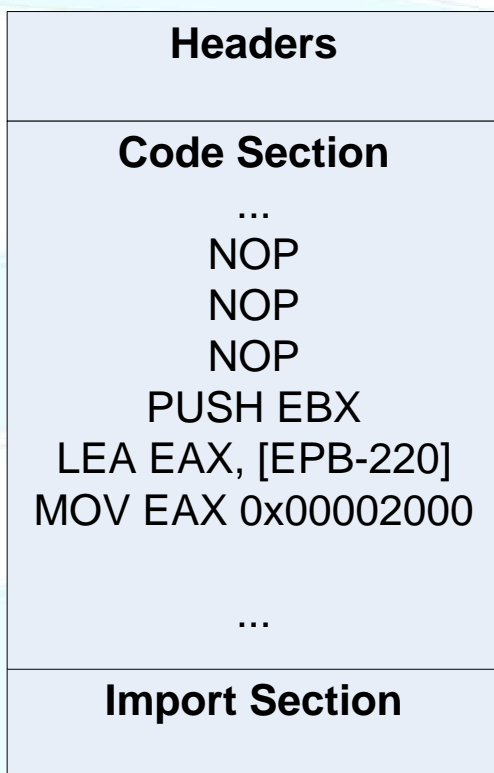
- **Code verification**
 - Code sections are read only in all modern OSES
 - Programs should not modify their own code
 - Check to see if the files on disk match what is running in memory

- **E.g.**
 - System Virginty Verifier (SVV)
 - VICE

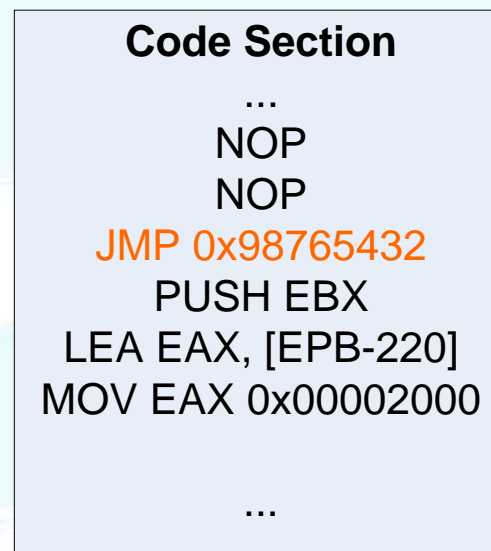


Detection Methodologies: Code Verification

MyApplication.exe
(on disk)



MyApplication.exe
(in memory)



Detection Methodologies

- **Cross View Detection**

- Take a view of a system at a high level. e.g. Windows Explorer
- Take a view of the system at a low (trusted) level. e.g. Raw Disk
- Registry, Files, Processes
- Compare the two

- **Examples**

- Sysinternals - Rootkit Revealer
- Microsoft Research – Strider Ghostbuster

- **Problems**

- What if someone hooks below your “trusted” level



Good Tools

- **Sysinternals**
 - Autoruns
 - Procexp
 - Rootkit Revealer
- **Icesword**
- **F-Secure Blacklight**
- **System Virginty Verifier**
- **Dark Spy**
- **RKDetector**



Detection Demo

- **Detecting the Hacker Defender rootkit**
 - **Rootkit Revealer**
 - **Icesword**





- [Rootkit Revealer](#)
- [Icesword](#)



Mitigation

- **Don't let an attacker get system level access... EVER!**
- **Host Intrusion Prevention**
- **Up to date antivirus and spyware protections**
- **Utilise the operating system tools available**
 - Windows XP SP2 DEP
 - Vista new technology



So where else can attackers hide?

- Hardware based rootkits
- Yes, they do exist in the wild!



Hardware Rootkits

- A OS reinstall won't save you
- Hard to remove.
 - Device is usually destroyed
- **Difficult to implement**
- **Very hard to detect**
- **With more and more memory on devices they are becoming prevalent with time**

- **VideoCardKit (<http://www.rootkit.com>)**
 - Stores code in FLASH or EEPROM
- **EEye Bootroot**
 - Installs in real mode via network PXE boot



Conclusions

- **Rootkit technology is an arms race**
- **Hard to tell who is winning**
- **Staying on top of developments is difficult**
- **Antivirus may catch up (one day...)**
- **Vista may solve some problems**
- **Firewalls do not provide protection**



Conclusions

- **No one tool will detect all rootkits, run at least 3 tools**
 - Rootkit Revealer
 - Fsecure Blacklight
 - IceSword
 - System Virginity Verifier
 - An updated Antivirus
- **It is impossible to re-establish trust on a compromised system**
- **If you are targeted with a custom rootkit you have very little chance of detecting it**
- **Network segregation, least privilege, internal host hardening all become extremely important**



Resources

- **Windows System Internals 4th Edition**– D. Solomon, M. Russinovich
- **Rootkits** – G. Hoglund, J. Butler
- **Primary Windows Rootkit Resource**
<http://www.rootkit.com>
- **Joanna Rutkowska – Stealth Malware Detection**
<http://www.invisiblethings.org>
- **F-Secure Blacklight**
 - <http://www.f-secure.com/blacklight/>
- **Sysinternals**
 - <http://www.sysinternals.com>
- **Rootkit Detector**
 - <http://www.rootkitdetector.com/>



Questions ?

<http://www.security-assessment.com>

darren.bilby@security-assessment.com

