ROOTKITS
ANALYSIS AND DETECTION

By
Jayanta Parial & Mukesh Kumar Singh

CERT-In
Ministry Of Communication & IT, New Delhi
Rootkit

The name, root kit, suggests a component that allows obtaining root access in a computer system, its only purpose is to help an attacker into keeping a previously obtained root access.
DEFINITIONS

• A collection of tools (programs) that a hacker uses to mask intrusion and obtain administrator-level access to a computer or computer network. ”

Courtesy: SANS
DEFINITIONS

• A hacker security tool that captures passwords and message traffic to and from a computer. A collection of tools that allows a hacker to provide a backdoor into a system, collect information on other systems on the network, mask the fact that the system is compromised, and much more. Rootkit is a classic example of Trojan Horse software. Rootkit is available for a wide range of operating systems. Courtesy: NSA
What does a Root Kit do?

• Hide Attacker Activities: Files, Processes and network connections
• Provide Unauthorized access
• Eavesdropping tools
• Clean Logs
• Hacking Tools
• Integrity Checkers deceivers
CLASSIFICATION

• Linux Root Kit
  – User Mode
  – Kernel Mode

• Windows Root Kit
  – Kernel Mode
USER MODE ROOTKIT

• Replace specific system program used to extract information from the system
• Can include additional tools like sniffers and password crackers

Files usually substituted:
• File Hiding: du, find, sync, ls, df, lsof, netstat
• Hide PROCESSES: killall, pidof, ps, top, lsof
• SNIFFING & data acquisitions: ifconfig (hide the PROMISC flag), passwd
Files usually substituted:

- Hide CONNECTIONS: netstat, tcpd, lsof, route, arp
- Execute tasks: crontab, reboot, halt, shutdown
- Hide LOGS: syslogd, tcpd
- Hide LOGINS: w, who, last. . . (no recording in utmp, wtmp, btmp, lastlog. . . )
- BACKDOORS: inetd, login, rlogin, rshd, telnetd, sshd, su, chfn, passwd, chsh, sudo
Tools to Hide evidence

- addlen: tool to fit the trojaned file size to the original one.
- fix: changes the creation date and checksum (non-cryptographic) of any program.
- wted: has edit capabilities of wtmp and utmp log files.
- zap: zeroes out log files (utmp, wtmp, lastlog (Solaris), messages. . . ) entries.
- zap2 (z2): erases log files entries: utmp, wtmp, lastlog. . .
Disadvantages

– Too many binaries to replace thus prone to mistakes
– Verifications through checksums is easy and OS dependent.

Some Famous Root Kits

– *T0rnkit:*
– *LRK, The Linux Rootkit:*
  » There are many others coming up every day.
KERNEL MODE ROOT KIT

• User mode root kit requires various binaries to be manipulated, Kernel mode requires only altering the kernel
• The kernel rootkits provide all the user-mode rootkit features from a low level, and their hiding and deceive capabilities can trick all user-mode inspection tools.
• The goal of a kernel rootkit is placing the malicious code inside the kernel source by manipulating the kernel.
INTERCEPTING EXECUTION FLOW

- Process
- System library
  - getdents()
- User mode
- System call interface
  - sys_getdents()
  - (manipulated)
- Kernel mode
- Kernel functions
Kernel mode →

- Choose interrupt handler → Interrupt Descriptor Table
- Choose system call → Syscall Table

sys_getdents()

- Access virtual filesystem
- Access actual filesystem...

© CERT-IN 29th March 2005
Kernel mode

choose interrupt handler

choose system call
(modified)

Interrupt Descriptor Table

Syscall Table
(modified copy)

Rootkit

sys_getdents()

access virtual filesystem

access actual filesystem

...
Kernel mode

choose interrupt handler

choose system call

Interrupt Descriptor Table

Syscall Table

Rootkit

sys_getdents()

access virtual filesystem

access actual filesystem

...
ROOTKIT DETECTION

• Anomaly Search
  – Files
  – Network Usage
  – Scheduled and Booting Tasks
  – Accounts
  – Log and User History entries
ROOTKIT DETECTION

• /proc psuedo file system
  – /proc/cmdline
  – /proc/kcore
  – /proc/kmsg
  – /proc/ksyms
  – /proc/modules
  – /proc/version/proc/sys
ROOTKIT DETECTION

• Suspicious files, directories and disk usage
  – System files in /tmp, /dev, font directories
  – Hard link count and directory size
  – Hard Link Count Analysis
  – Total Block Count Analysis

• MAC Times
  – Time Stamp Analysis
ROOTKIT DETECTION

• Logging system call traces: strace
• Detecting (and recovering) deleted executables and open files
• Network Connections
• Detecting Promiscuous NIC
• Integrity
• Checking Rootkit features
ROOTKIT DETECTION

• Tools
  – Saint Jude
  – Chrootkit
  – Rootkithunter
  – RkScan
  – The “Carbonite” LKM
  – Kstat
  – Exporting standard and debugging module symbols
  – Kernel memory scanning:
  – System Call table help:LKM or memory dump
  – Execution path analysis
  – CheckIDT
  – The kern_check tool
  – The check_ps tool
PROTECTING LINUX KERNEL

• OS Hardening
• Patching the kernel vulnerabilities
• Linux Bootstrap process analysis
• Kernel compilation without module support
• Kernel Hardening
• Restricted operations and capabilities
• “System.map” Protection
• System call table export
PROTECTING LINUX KERNEL

• LKM Protection
  – modlock (LKM Locking)
  – syscall_sentry LKM
  – Toby LKM
  – St. Michael;
  – LIDS