Andrew Parker, Alex Cater
CS 493
Dr. Lawlor
12/11/19

Project Writeup

For our project, we have decided to write a linux kernel module capable of providing root, along with some other basic functionalities. Initially our project plan was to infect a VM with a rootkit and analyze the effects of the rootkit. However, we changed our project to implementing our own rootkit and analyzing its effects. The main purpose of this project was to gain experience working with linux under the hood through writing a kernel module, as well as to gain a basic understanding of how to go about writing a rootkit with some functionalities. During our process, at some points we used code online to help us with our implementation. Credits are given at the end of this document.

Our kernel module creates a character device that we able to define arbitrary functions to be used when a user tries to write or read data from the device. When a user writes the correct "password" to the device then the module gets the user's current user id and changes it to 0, giving them root access. Since we were able to define the functions to be any arbitrary code, we could extend our rootkit capabilities, such as adding a keylogger.

When we write 'keylogger' to the device we created, our keylogger is turned on. This is displayed in the kernel logs. During initialization, the keylogger creates a file to write the keys pressed to. The file can be accessed in /sys/kernel/debug/lkmr/keys. We have a keyboard notifier block that is calling code whenever a keyboard event happens. This code translates the given

scan codes that were pressed to readable keys. This is written to a string buffer that writes to the file ‘keys’.

Overall, we learned a lot from this project. We gained experience writing our first kernel module that worked on a lower level of the operating system. We also learned about working with devices, as well as the innerworkings of rootkits and keyloggers.

Functionality

- prints to kernel logs
- creates a device ``/dev/ttyR0``
    - has custom read/write/open/release functions
    - “echo ‘root’ >> /dev/ttyR0” gives root access to user
    - able to read string stored in kernel using a userspace program
    - “echo ‘keylogger’ >> /dev/ttyR0” turns on the keylogger. Keys pressed are written to the file ‘keys’ located in /sys/kernel/debug/lkmr.
    - “echo ‘exitkeylogger’ >> /dev/ttyR0” turns off the keylogger.

References

https://github.com/jarun/keysniffer

https://0x00sec.org/t/kernel-rootkits-getting-your-hands-dirty/1485