Programming Homework: Rootkits

Task 1: Kernel-level rootkit

Root.c creates a character device that allows the user to get root privileges on whatever machine it is installed by writing the string “g0tR0ot” to the device file.

The device name is defined in line 14 as "ttyR0" and its class name is defined in line 15 as "ttyR.” This results in the creation of a device at /sys/class/ttyR/ttyR0.

- The file operations (fops) structure is defined, which specifies the callback functions for various operations that can be performed on the character device, such as opening, reading, and writing.
- root_open() is called every time the device is opened.
- root_read() returns the length of the buffer, indicating that the read operation is successful.
- root_write() allocates memory to store the incoming data in line 68 and compares it with a predefined magic string ("g0tR0ot") in line 73. If the comparison is successful, it gives the user root privileges using the prepare_creds() and commit_creds() functions.
- root_init() is the entry point for the module. It registers the character device with the kernel using the register_chrdev() function, creates a device class using the class_create() function, and creates a device under that class using the device_create() function.
- root_exit() is the exit point for the module. It destroys the character device, unregisters the device class, and unregisters the character device using the functions device_destroy(), class_unregister(), class_destroy(), and unregister_chrdev().

Task 2: What else?

Reptile

1. Root access: Gives root to unprivileged users by typing /reptile/reptile_cmd root.
2. Hide files and directories: All files and folders that have reptile in the name will be hidden
3. Hide processes: Hides processes using /reptile/reptile_cmd hide <pid> and show processes using /reptile/reptile_cmd show <pid>.
4. Hide TCP/UDP connections: Hides connections using /reptile/reptile_cmd conn <IP> hide and show connections using /reptile/reptile_cmd conn <IP> show.
5. File content tampering: All content in between the tag #<reptile> will be hidden.
6. Heaven's door: A ICMP/UDP port-knocking backdoor

NRootkit

7. Keyboard listener: Captures keystrokes of users.

Flame

8. Audio recording: Captures environment sounds via the system's microphone,
9. Screen recording: Grabs and stores frequent screenshots of activity on the machine.
10. Sends information: Forwards saved information to a remote server.
11. Bluetooth beacon: Scans for other bluetooth-enabled devices in the vicinity and siphons names and phone numbers from their contacts folder.

12. Sniffer: Scans all of the traffic on a machine's local network and collect usernames and password hashes that are transmitted across the network.
13. USB infection: Can spread via USB using autorun and .lnk vulnerabilities.

Zeus

14. Information stealing: Uses keylogging or form-grabbing to get usernames and passwords.
15. Build a botnet: Maintains contact with its operator through a command-and-control (C&C) server so that it can remotely receive additional instruction.

ZeroAccess

16. Click fraud: Turns the host machine into a botnet and simulates clicks on website advertisements paid for on a pay per click basis.
17. Bitcoin mining: Turns the host machine into a botnet and uses the CPU to mine for bitcoin.

Stuxnet

18. Mutation: Is metamorphic in order to hide detection.
19. PLC altering: Alters PLCs used for controlling uranium centrifuges to damage machinery.
20. Create false data: Sends false data to the PLC so that incorrectly functioning centrifuges remain undetected.