CS5231 System Security Homework 3

Lee Kai Wen, Aloysius (A0154597N)

Task 1

Configuration for auditd

The following line is added to */etc/auditbeat/audit.rules.d/audit-rules.conf* to capture the *open*, *openat*, *read*, *write* and *writev* syscalls:

```
-a always,exit -S open,openat,read,write,writev
```

Analysis of Audit Logs

Design of Analysis Script

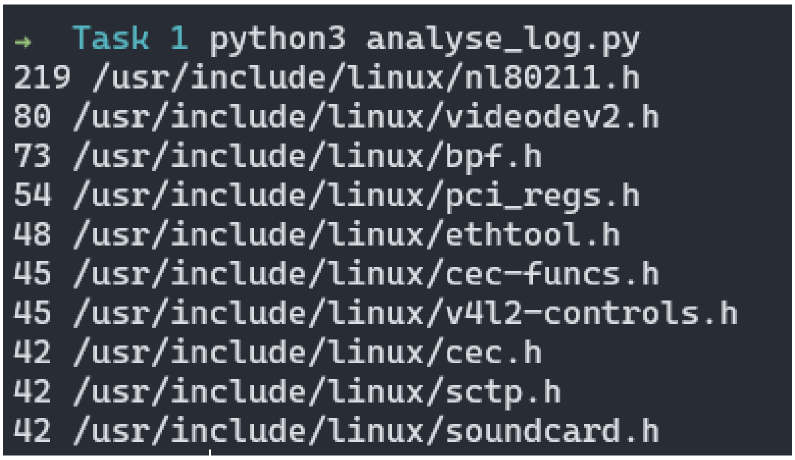
The analysis script *analyse_log.py* is developed in Python3 and performs the following actions:

1. Create a dictionary (hash map) with a default value of 0 for non-existent keys
2. Reads the parsed log file *task1_parsed.log* line by line
3. Remove the angle backets and split each line by the delimiter "," and retrieve the 4th item which is the file path
4. Check if the file path is in the directory */usr/include/linux* by checking if starts with *"/usr/include/linux"*
5. If it does, increment the counter for the file in the dictionary
6. After processing all the logs, sort the resulting dictionary of files in descending order accesses then alphabetically
7. Print the top 10 most accessed files

Running the Analysis Script

1. Ensure the parsed log file is in the same directory as analyse_log.py.
2. Modify the variable `PARSED_LOG_FILE` to the name of the parsed log file. The default is `task1_parsed.log`.
3. Run the analysis script with: `python3 analyse_log.py`

Analysis Script Output



```
→  Task 1 python3 analyse_log.py
219 /usr/include/linux/nl80211.h
80 /usr/include/linux/videodev2.h
73 /usr/include/linux/bpf.h
54 /usr/include/linux/pci_regs.h
48 /usr/include/linux/ethtool.h
45 /usr/include/linux/cec-funcs.h
45 /usr/include/linux/v4l2-controls.h
42 /usr/include/linux/cec.h
42 /usr/include/linux/sctp.h
42 /usr/include/linux/soundcard.h
```

Top Ten Most Accessed Files

The top ten most accessed files under the directory */usr/include/linux* are:

S/N	Times Accessed	File
1	219	/usr/include/linux/nl80211.h
2	80	/usr/include/linux/videodev2.h
3	73	/usr/include/linux/bpf.h
4	54	/usr/include/linux/pci_regs.h
5	48	/usr/include/linux/ethtool.h
6	45	/usr/include/linux/cec-funcs.h
7	45	/usr/include/linux/v4l2-controls.h
8	42	/usr/include/linux/cec.h
9	42	/usr/include/linux/sctp.h
10	42	/usr/include/linux/soundcard.h

Task 2

Pseudocode

The pseudocode for *cs5231_file_permission* checks if *malicious_prog* is accessing the sensitive files. Access to the file denied when the current process is *malicious_prog* and the current file being opened are the sensitive files */usr/include/linux/if.h* or */usr/include/linux/u.h*, otherwise access is allowed.

```
function cs5231_file_permission(file, mask) {
  // Process information
  cur_task = get_current_task()
  process_name = cur_task.name

  // File information
  file_path = file.path

  isMaliciousProg = process_name == "malicious_prog"
  isIfH = file_path == "/usr/include/linux/if.h"
  isUnH = file_path == "/usr/include/linux/un.h"
  isSensitiveFile = isIfH or isUnH

  // Check if malicious program is reading sensitive files
  if (isMaliciousProg and isSensitiveFile) {
    print("Sensitive file {file_path} access is detected.")

    // Denied access to file
    return -EACCES
  }

  // Allowed access to file
  return 0
}
```