Task 2. Анализ дампа памяти с помощью утилиты Volatility

1. Смотрим историю bash

```
vagrant@kali:~$ volatility --profile=LinuxUbuntu16_04_4_4_0_116_genericx64 --filename=/vagrant/task2/image linux_bash
Volatility Foundation Volatility Framework 2.6
Pid      Name                 Command Time                   Command
-------- -------------------- ------------------------------ -------
    1166 bash                 2018-04-15 15:24:47 UTC+0000   cd ..
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls
    1166 bash                 2018-04-15 15:24:47 UTC+0000   sudo rm /media/sf_bin/
    1166 bash                 2018-04-15 15:24:47 UTC+0000   cd .cache/
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls
    1166 bash                 2018-04-15 15:24:47 UTC+0000   sudo su
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:24:47 UTC+0000   poweroff
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:24:47 UTC+0000   sudo chown panda:panda ht0p
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls
    1166 bash                 2018-04-15 15:24:47 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:24:47 UTC+0000   sudo umount /media/sf_bin
    1166 bash                 2018-04-15 15:24:47 UTC+0000   sudo rm -r /media/sf_bin/
    1166 bash                 2018-04-15 15:24:47 UTC+0000   chown panda:panda ht0p
    1166 bash                 2018-04-15 15:24:47 UTC+0000   chown panda:panda suleanu
    1166 bash                 2018-04-15 15:24:47 UTC+0000   mv suleanu ht0p
    1166 bash                 2018-04-15 15:24:49 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:24:55 UTC+0000   shred -u .bash_history
    1166 bash                 2018-04-15 15:24:59 UTC+0000   ls
    1166 bash                 2018-04-15 15:25:02 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:25:21 UTC+0000   ls /media/
    1166 bash                 2018-04-15 15:25:24 UTC+0000   ls -la
    1166 bash                 2018-04-15 15:25:30 UTC+0000   ./ht0p \  &
    1166 bash                 2018-04-15 15:25:32 UTC+0000   htop
    1166 bash                 2018-04-15 15:25:32 UTC+0000   htop
```

из истории видно, что был запущен файл ht0p запущенный в бэкграунде. Этот файл был скопирован с какого-то устройства, еще видна попытка почистить bash_history

2. Командой linux_pstree смотрим дерево процессов

```
.iscsid                1036
.iscsid                1037
.irqbalance            1079
.login                 1084
..bash                 1166              1000
...ht0p                1192              1000
...htop                1193              1000
.systemd               1157              1000
..(sd-pam)             1160              1000
[kthreadd]             2
```

видим что ht0p не порождал другие процессы и имеет pid **1192**

```
0x0000000035597000 jbd2/sda1-8         297    -    -1    -1    0xf000f84dc0000022 -
0x00000000355e0000 sleep               1100   -    -1    -1    0xf000f84dc0000022 -
0x00000000355e0e00 update-motd-fsc     1140   -    -1    -1    0xf000f84dc0000022 -
0x00000000355e1c00 update-motd-fsc     1150   -    -1    -1    0xf000f84dc0000022 -
0x00000000355e2a00 update-motd-fsc     1144   -    -1    -1    0xf000f84dc0000022 -
0x00000000355e3800 ht0p                1192   -    -1    -1    ------------------ -
0x00000000355e4600 update-motd-fsc     1136   -    -1    -1    0xf000f84dc0000022 -
0x00000000355e5400 kpsmoused           139    -    -1    -1    0xf000f84dc0000022 -
0x00000000355e6200 htop                1193   -    -1    -1    ------------------ -
0x00000000355e7000 dumpe2fs            1135   -    -1    -1    0xf000f84dc0000022 -
0x0000000035608000 find                589    -    -1    -1    0xf000f84dc0000022 -
0x0000000035608e00 wc                  590    -    -1    -1    0xf000f84dc0000022 -
```

3. Командой linux_getcwd ищем путь откуда был запущен скрипт.

```
vagrant@kali:~$ volatility --profile=LinuxUbuntu16_04_4_4
 | grep 1192
Volatility Foundation Volatility Framework 2.6
ht0p                 1192 /home/panda
```

4. Командой linux_find_file узнаем Inode.

```
vagrant@kali:~$ volatility --profile=LinuxUbuntu16_04_4_4_0_116_genericx64 -
ile -F "/home/panda/ht0p"
Volatility Foundation Volatility Framework 2.6
Inode Number                 Inode File Path
---------------- ------------------ ---------
          390593 0xffff88007bd8e698 /home/panda/ht0p
```

вытаскиваем в файл

```
vagrant@kali:~$ volatility --profile=LinuxUbuntu16_04_4_4_0_116_genericx64 --filename=/vagrant/task2/image linux_find_f
ile -i 0xffff88007bd8e698 -O /vagrant/ht0p_report
Volatility Foundation Volatility Framework 2.6
```

5. Открытие файла командой xxd и hexdump ничего не дает.