CSEC 730 - Advanced Computer Forensics
Lab 1 - Linux Forensic Analysis

Please submit your answers (in PDF format) to the assignment submission folder under *myCourses* > *Assignments* by the deadline.

Objective

This lab will use *Sleuthkit* and *Autopsy (GUI)* to analyze a Linux image. You will practice the Sleuthkit tools at the data layer, metadata layer, file system layer, and file name layer in **part 1**. In **part 2** and **part 3**, you will learn Autopsy, a GUI-based front-end for Sleuthkit's basic features for forensic analysis. Finally, you will provide a short conclusion in **part 4** for your analysis of a given image. The steps provided below are only the guidelines. Please feel free to try a variety of Sleuthkit tools with different options to fully understand this powerful tool.

Case Scenario

Mark Watson works as a Director of Finance at an advertising firm. He suspects that a contractor, Frank Lewis, has read the confidential annual financial report (*Earnings.xls*) to influence his next contract with the firm. The IT administrator informed you that there is a Linux-based file server in the office where all employees save the official documents. Mark and Frank each have their own folders on this server.
You have been given the image of the hard drive, *Linux_Financial_Case.001*, to find any evidence that suggests Frank may have read Earnings.xls.

Evidence File

The acquired image "*Linux_Financial_Case.001.zip" is posted on myCourses* > *Content* > *Hands-on Labs* > *Lab 1*.

After you download and extract the image .zip file, validate both its md5 and sha1 hash values:

- MD5 (Linux Financial Case.001) = 7b39de0ca146c89ad73d1d421c8f7a05
- SHA1 (Linux Financial Case.001) = c7b06f006ff79711e692bd2620aba4cc2a4426d2

Deliverable

Answer all the exercise questions and include screenshots as supporting data if required.

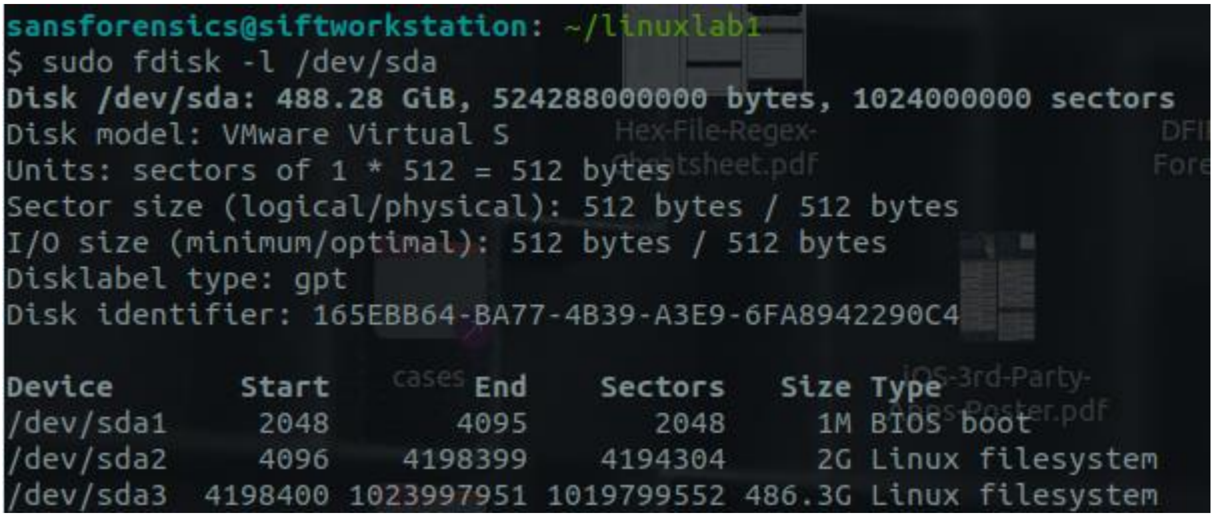
****

PART 1. Practice the sleuthkit command line tools to analyze the image “*Linux_Financial_Case.001”* (Each question is 2.5 points)

Instructions

1. Launch the SIFT Workstation VM. The default login username is **sansforensics**, and the default password is **forensics.**

   Question 1. How many partitions do your SIFT VM’s */dev/sda* have? What is the offset of the starting sector for the “Linux” partition? Show the commands and screenshots.

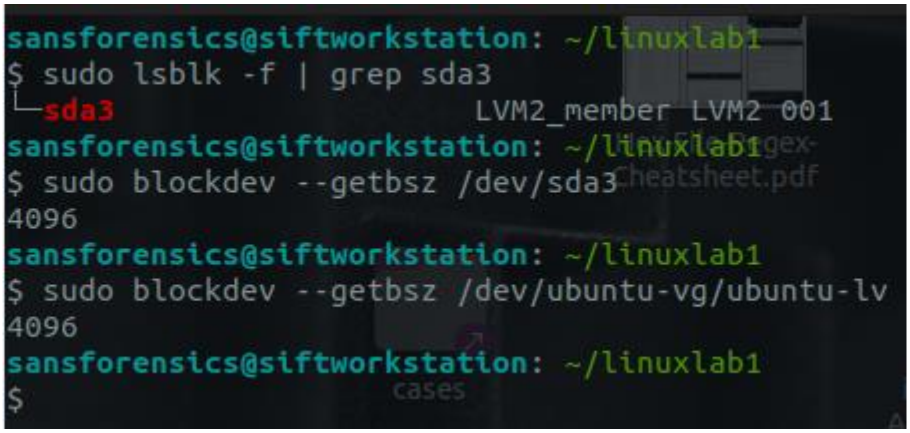


```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo fdisk -l /dev/sda
Disk /dev/sda: 488.28 GiB, 524288000000 bytes, 1024000000 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 165EBB64-BA77-4B39-A3E9-6FA8942290C4

Device        Start        End    Sectors   Size Type
/dev/sda1      2048       4095       2048     1M BIOS boot
/dev/sda2      4096    4198399    4194304     2G Linux filesystem
/dev/sda3   4198400 1023997951 1019799552 486.3G Linux filesystem
```

The offset sector for Linux is 4198400.

   Question 2. Which file system does this “Linux” partition use? What is the block size of this “Linux” partition? Show the commands and screenshots.



```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo lsblk -f | grep sda3
└─sda3                       LVM2_member LVM2 001
sansforensics@siftworkstation: ~/linuxlab1
$ sudo blockdev --getbsz /dev/sda3
4096
sansforensics@siftworkstation: ~/linuxlab1
$ sudo blockdev --getbsz /dev/ubuntu-vg/ubuntu-lv
4096
sansforensics@siftworkstation: ~/linuxlab1
$
```

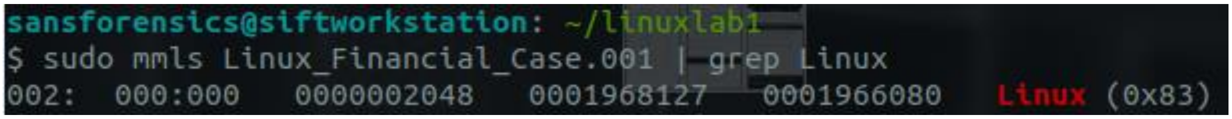
The file System: LVM

The Block Size: 4096

The rest of the questions are related to the image "Linux_Financial_Case.001"

2. The image "Linux_Financial_Case.001" contains one partition. In able to analyze this image, you have to first find the offset of the starting sector for the partition.

   Question 3. What is the command along with the appropriate options you used?



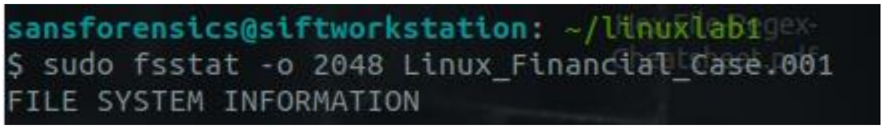
```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo mmls Linux_Financial_Case.001 | grep Linux
002:  000:000   0000002048   0001968127   0001966080   Linux (0x83)
```

   The Command: sudo mmls Linux_Financial_Case.001 | grep Linux

   ____

3. Find the image's file system information (hint: you have to provide the offset you got from step 2.)

   Question 4. What is the command along with the appropriate options you used to find the file system?



```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo fsstat -o 2048 Linux_Financial_Case.001
FILE SYSTEM INFORMATION
```

   The Command: sudo fsstat -o 2048 Linux_Financial_Case.001

   ____

   Question 5. What type of file system is the image used? **Ext2**

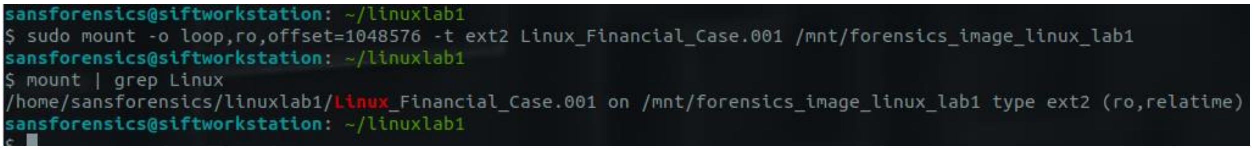
   ____

   Question 6. In what scenarios, you do NOT have to use the offset option –o for a sleuthkit command?

   When the image contains only one file system starting at the beginning of the image.

   ____

   Question 7. Provide the options of *mount* you will run to mount the Linux_Financial_Case.001 image's partition for forensics investigation? (Show a screenshot of the mounted filesystem)



```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo mount -o loop,ro,offset=1048576 -t ext2 Linux_Financial_Case.001 /mnt/forensics_image_linux_lab1
sansforensics@siftworkstation: ~/linuxlab1
$ mount | grep Linux
/home/sansforensics/linuxlab1/Linux_Financial_Case.001 on /mnt/forensics_image_linux_lab1 type ext2 (ro,relatime)
sansforensics@siftworkstation: ~/linuxlab1
$
```

   The Command: $ sudo mount -o loop,ro,offset=1048576 -t ext2 Linux_Financial_Case.001 /mnt/forensics_image_linux_lab1

4. Use *fls* to list the deleted files and directories, as a mactime body (-m), and save the file as *flsBody*.

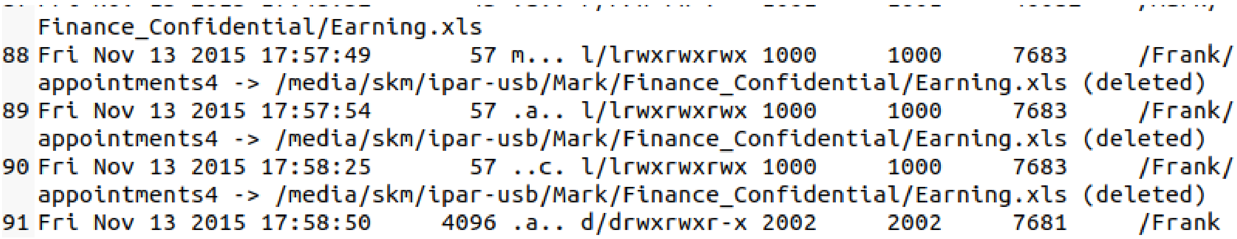
   Question 8. What is the command along with appropriate options you used?

   fls -f ext2 -m / -r -o 2048 Linux_Financial_Case.001 > flsBody

   ____

5. Use Sleuthkit's *mactime* to create a timeline of *flsBody*. Save the timeline in a file called *flsMactime* and examine the timeline.

   Question 9. What is the command along with the appropriate options you used? Include a screenshot of a part of the content of *flsMactime*.



```
Finance_Confidential/Earning.xls
Fri Nov 13 2015 17:57:49       57 m... l/lrwxrwxrwx 1000     1000     7683     /Frank/
appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)
Fri Nov 13 2015 17:57:54       57 .a.. l/lrwxrwxrwx 1000     1000     7683     /Frank/
appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)
Fri Nov 13 2015 17:58:25       57 ..c. l/lrwxrwxrwx 1000     1000     7683     /Frank/
appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)
Fri Nov 13 2015 17:58:50     4096 .a.. d/drwxrwxr-x 2002     2002     7681     /Frank
```

   The Command: mactime -b flsBody > flsMacTime

   ____

6. Use *ils* to list the inode information for all deleted files, as a mactime body (-m), and save the file as *ilsBody*.

   Question 10. What is the command along with the appropriate options you used?

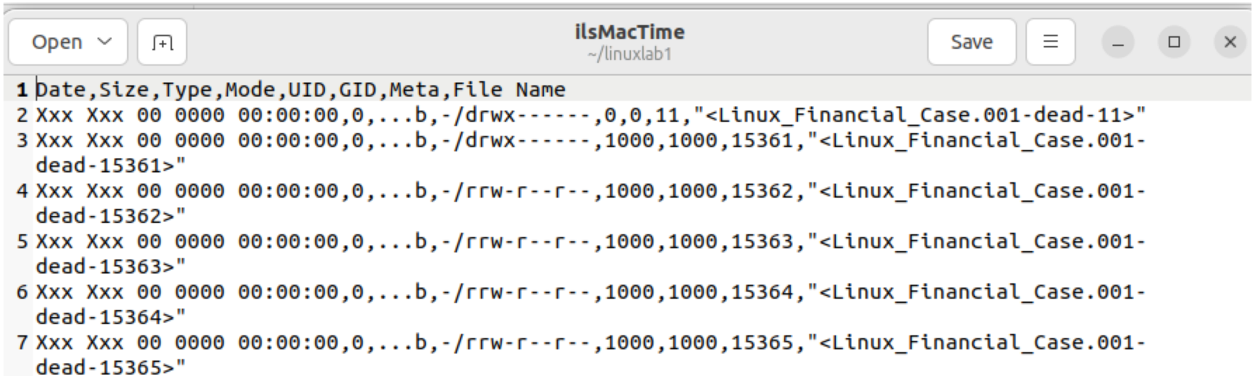
   The Command: sudo ils -m -o 2048 Linux_Financial_Case.001 > ilsBody

   ____

7. Use Sleuthkit's *mactime* to create a timeline of *ilsBody*. Save the timeline in a file called *ilsMactime* and examine the timeline.

   Question 11. What is the command along with appropriate options you used? Include a screenshot of a part of the content of *ilsMactime*.

   The Command: mactime -b ilsBody -d > ilsMactime



```
Date,Size,Type,Mode,UID,GID,Meta,File Name
Xxx Xxx 00 0000 00:00:00,0,...b,-/drwx------,0,0,11,"<Linux_Financial_Case.001-dead-11>"
Xxx Xxx 00 0000 00:00:00,0,...b,-/drwx------,1000,1000,15361,"<Linux_Financial_Case.001-
dead-15361>"
Xxx Xxx 00 0000 00:00:00,0,...b,-/rrw-r--r--,1000,1000,15362,"<Linux_Financial_Case.001-
dead-15362>"
Xxx Xxx 00 0000 00:00:00,0,...b,-/rrw-r--r--,1000,1000,15363,"<Linux_Financial_Case.001-
dead-15363>"
Xxx Xxx 00 0000 00:00:00,0,...b,-/rrw-r--r--,1000,1000,15364,"<Linux_Financial_Case.001-
dead-15364>"
Xxx Xxx 00 0000 00:00:00,0,...b,-/rrw-r--r--,1000,1000,15365,"<Linux_Financial_Case.001-
dead-15365>"
```

8. Compare the number of entries from *ilsMactime* and from *flsMactime*.

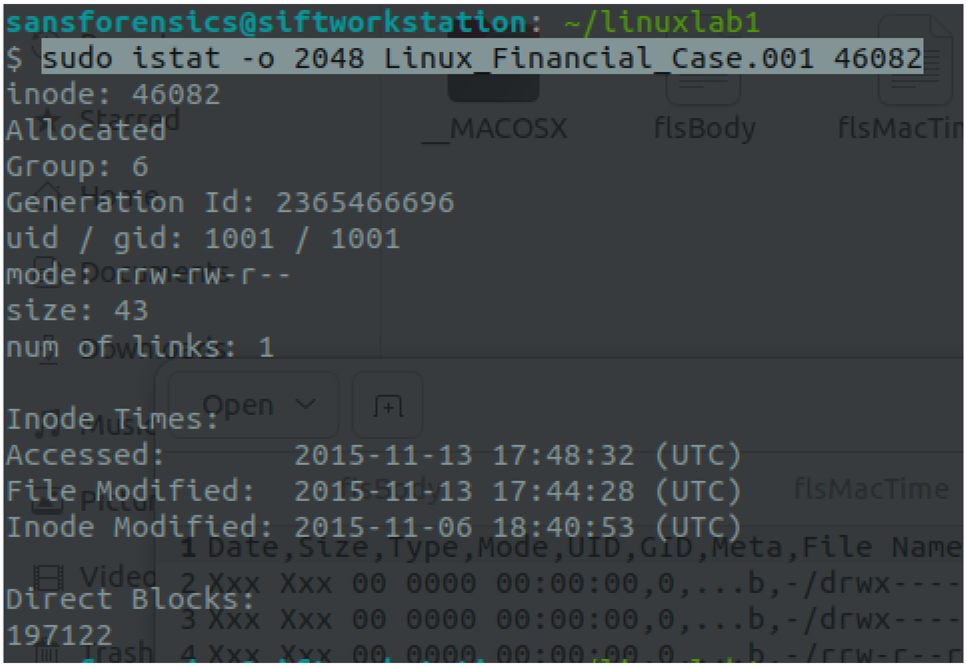
Question 12. Do *ilsMactime* and *flsMactme* have the same number of entries? Explain your findings.

I believe that the ils would be focus on the inodes, which include both allocated and unallocated inodes(deleted files). On the other hand, fls lists both active and deleted files and directories within the file system.

9. use *istat* to view the details of the inode 46082.

Question 13. What is the command along with appropriate options you used? Include a screenshot.

The Command: sudo istat -o 2048 Linux_Financial_Case.001 46082



```
sansforensics@siftworkstation: ~/linuxlab1
$ sudo istat -o 2048 Linux_Financial_Case.001 46082
inode: 46082
Allocated
Group: 6
Generation Id: 2365466696
uid / gid: 1001 / 1001
mode: rrw-rw-r--
size: 43
num of links: 1

Inode Times:
Accessed:       2015-11-13 17:48:32 (UTC)
File Modified:  2015-11-13 17:44:28 (UTC)
Inode Modified: 2015-11-06 18:40:53 (UTC)

Direct Blocks:
197122
```

__MACOSX
flsBody
flsMacTi
Open
flsMacTime
1 Date,Size,Type,Mode,UID,GID,Meta,File Name
2 Xxx Xxx 00 0000 00:00:00,0,...b,-/drwx----
3 Xxx Xxx 00 0000 00:00:00,0,...b,-/drwx----

10. use *icat* to dump out data from the inode 46082.

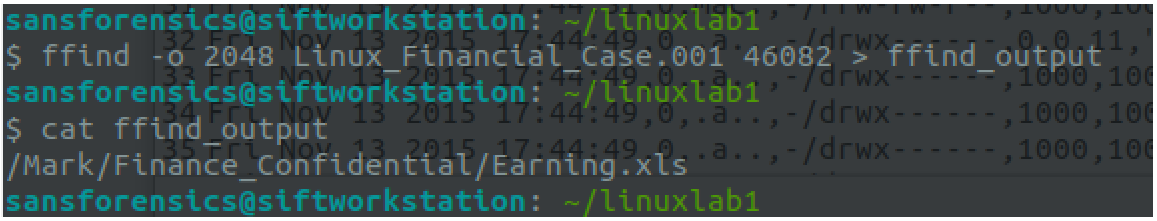
Question 14. What is the command along with appropriate options you used?

The Command: sudo icat -o 2048 Linux_Financial_Case.001 46082 > icat_output

11. Use *ffind* to find the file's filename that has the inode 46082.

Question 15. What is the command along with appropriate options you used? Include a screenshot.

The Command: ffind -o 2048 Linux_Financial_Case.001 46082 > ffind_output



```
sansforensics@siftworkstation: ~/linuxlab1
$ ffind -o 2048 Linux_Financial_Case.001 46082 > ffind_output
sansforensics@siftworkstation: ~/linuxlab1
$ cat ffind_output
/Mark/Finance_Confidential/Earning.xls
sansforensics@siftworkstation: ~/linuxlab1
```

12. Use *blkcat* to dump out the data content of the datablock 197122

Question 16. What is the command along with the appropriate options you used?

The Command: blkcat -o 2048 Linux_Financial_Case.001 197122 > blkcat_output

Question 17. If a file with the inode 100 uses two block addresses, *block 1000* and *block 1001*, will "icat -f ext2 image 100" dump out the same content as the command "blkcat –f ext2 image 1000"? Explain your answer.

No, icat dumps the entire content of the file, while blkcat dumps the content of a specific block.

13. Use *ifind* to find the inode number that one of its correspondent data blocks is 197122.

Question 18. What is the command along with appropriate options you used? Also provide a case scenario that shows the usefulness of *ifind*.

The Command: ifind -d 197122 -o 2048 Linux_Financial_Case.001

Case Scenario: In a cybersecurity incident response, an investigator is analyzing a compromised server to identify the extent of a data breach. During the investigation, a suspicious block number (197122) is discovered, which is suspected to contain malware or unauthorized data. Using the ifind command, the investigator can determine the inode number associated with this block. With this inode number, the investigator can then use additional forensic tools to examine the file's properties, content, and potential impact on the system. This information is crucial for understanding the nature of the breach, mitigating the threat, and preventing future incidents.

PART 2. Use Autopsy to analyze "*Linux_Financial_Case.001*" *case* (Each question is 2.5 points)

Instructions

To Start autopsy:
Start a terminal (go to applications -> Accessories->Terminal) and type in
$ sudo autopsy
While this process is running, open a web browser point it
to the URL indicated – http://localhost:9999/autopsy

Click on "New Case".

Enter "linux_financial_case" as the case name, you may fill in other optional information, then click "New Case". Confirm the information and click "OK". (Names with spaces will not work.)

Click "Add Host".
Enter "Host1" under "Host Name" and "EST" under "Timezone" and click "Add Host".

Confirm the information and click "ADD HOST".
Click "Add Image".
Click "ADD IMAGE FILE".

Select "Disk" since this image contains a disk image (vs a partition).

In "Location" type the path to the image file "*Linux_Financial_Case.001*".

Explore the various "Import Methods".

Review the options for checking/creating md5's and select the appropriate entry based on the information you currently have.

Question 1: Which option did you choose and why?
I choose Symlink option is that it allows for real-time analysis of the image file without the risk of corruption. Since a symlink is just a reference to the original file, any changes made to the original file are immediately reflected in the evidence locker.

Autopsy identifies the partition and the file system type of this partition.

Question 2: Which Sleuthkit tool does Autopsy use to display the partition table information?
I would say it is mmls

Question 3: Which Sleuthkit tool does Autopsy use to determine the file system type of this partition?
I would say it is fsstat

Click "Add" to add the image to host 1. And confirm the information and click "OK".

Now Autopsy should have mounted the partition.
Select the partition, click “Analysis” and choose “FILE ANALYSIS” tab.

In this mode, you can view file and directory metadata and file content.
Click the inode of directory *Mark*, 23041, to see the detail information about Mark’s directory.

Go to *Mark/Finance_Confidential* directory, and click on *Earning.xls* file. In the information window at the bottom, explore the “display”, “report”, “export” links.

Question 4 What information do you get from “display” and “report”? What does “export” do?
Display: View analysis results within Autopsy's interface.
Report: Generate a summary report of analysis findings.
Export: Save analysis data in different formats for further use or sharing.

From here you can recover any of the files shown, including deleted ones if the content has not been overwritten.

Question 5: How can you determine that a file has been deleted?
I would say there are three ways the first one is it marks as red color which indicates that it has been deleted, and the second one is that there is an item in the table of content which caked DEL which would give a check mark if the file is deleted, and lastly, is that there is an option called ALL DELETED FILES on the left side in the file analysis page which will list all the deleted files. Also, you can check oy going inside the file and it would say Not Allocated in the details or when you enter inside the file page you will the file path and it would say deleted next to it.

Click “File Type”. Then click “Sort Files by Type”. Then click “OK”.

Question 6: How is the “Sort Files by Type” formation useful in an investigation?
Sorting files by type in an investigation is useful for organizing evidence, identifying patterns, prioritizing analysis, and gaining a quick overview of the evidence.

To view the sorted file, click on “View Sorted Files” and copy/paste the URL into a browser.

Click on “Meta Data” and provide a valid inode number.

Question 7: Knowing an inode number, which Sleuthkit tool does Autopsy use to determine the data blocks referenced by the inode?
I would say it is istate

Click on the “Image Details” tab and read the information given.

Question 8: What information can you get from this window? Where does Autopsy get this information from?

Autopsy tells us that the disk image uses the Ext2 file system, commonly found in Linux. It shows the volume name and ID, which help identify the disk. Autopsy also gives us the last time the disk was written to, checked, and mounted, which can reveal recent activity and any issues with how it was unmounted. It tells us the file system's features, like extended attributes and resizable inodes, and gives details about the file and block ranges, block size, and free blocks. This helps us understand how the files are stored on the disk. Autopsy gets all this info by looking at the disk using The Sleuth Kit.

Click the “Close” tab to close the “Analyze”, and you will be back to the “Host Manager”.

Select the partition and click “File Activity Timelines”

Click “Create Data File”.
Select the disk partition and click “OK”, and confirm the information.

Question 9: What Sleuthkit command line tool(s) was/were used to generate the body file?

I believe the mactime command

Click “OK”.

Now we have the body file, we can sort the body file to generate a timeline.
Autopsy this version has a bug for creating a timeline. To fix it, you will run “sudo cp /usr/bin/mactime /usr/bin/mactime-sleuthkit”.

In the “Create Timeline” window, you can select the starting and ending dates of file activity that you want to see, for example, from Jan. 2015 to Jan. 2016.

Note the sorted information. Click the links at the top to look at other dates.

Question 10: How might this timeline information be useful for forensic investigations?

The timeline helps investigators see when files were created, accessed, or modified. It can reveal patterns or anomalies, corroborate statements, and focus the investigation on specific events or time frames.

Click “Close”.
Back to “Host Manager”

Explore any other features of Autopsy & Sleuthkit you would like to.

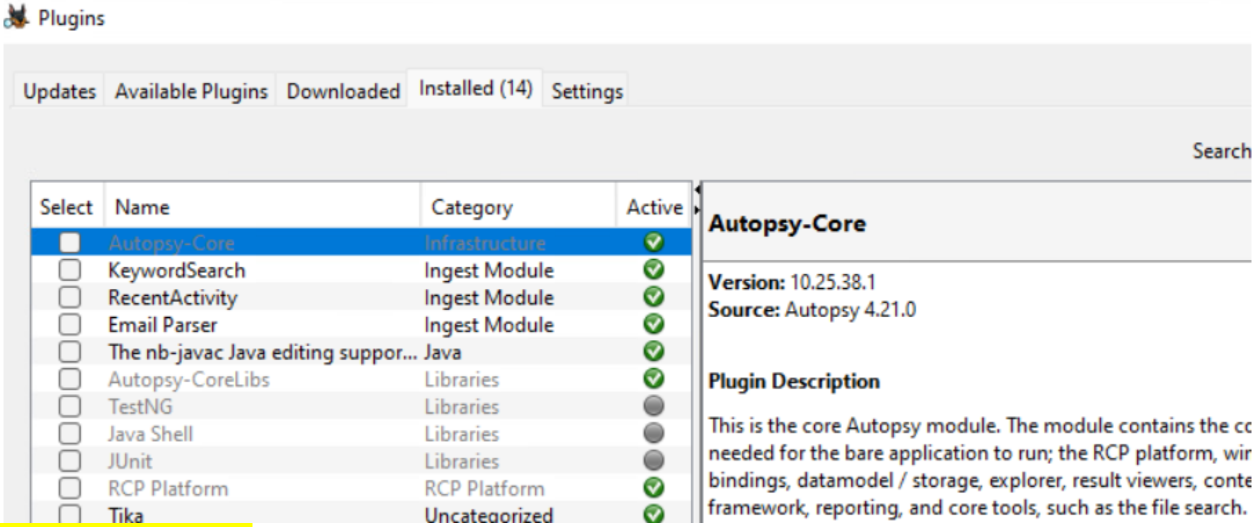
After you are done, close the case by clicking “Close Host” then “Close Case”. You can reopen the case to work on it later if you choose to.

Part 3. Use Windows Autopsy to analyze “*Linux_Financial_Case.001*” *case* (16 points)

Instruction: Download the latest Autopsy for Windows from https://www.autopsy.com/download/, install it on your Windows Forensic machine, and analyze “Linux_Financial_Case.001”.

1) List at least three features of Windows Autopsy with screenshots. (10 points)

1- Plugins



Select	Name	Category	Active
□	Autopsy-Core	Infrastructure	✓
□	KeywordSearch	Ingest Module	✓
□	RecentActivity	Ingest Module	✓
□	Email Parser	Ingest Module	✓
□	The nb-javac Java editing suppor...	Java	✓
□	Autopsy-CoreLibs	Libraries	✓
□	TestNG	Libraries	●
□	Java Shell	Libraries	●
□	JUnit	Libraries	●
□	RCP Platform	RCP Platform	✓
□	Tika	Uncategorized	✓

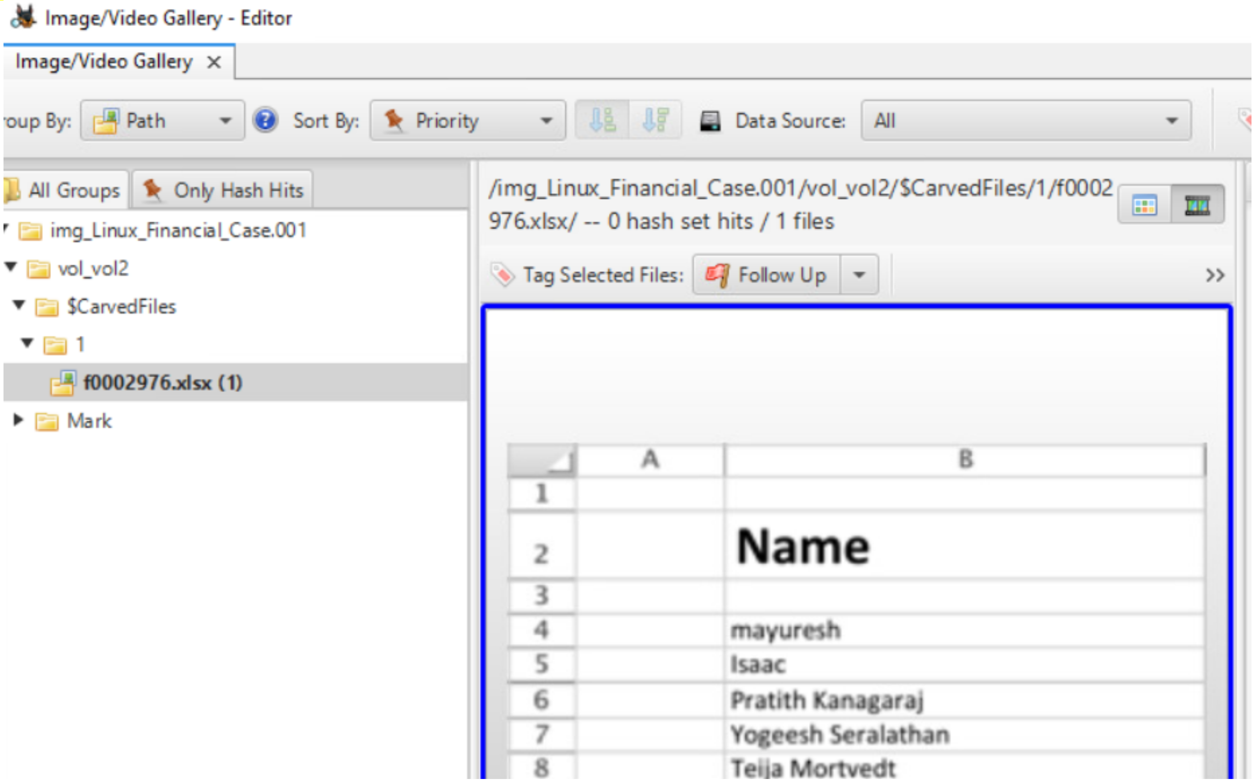
Autopsy-Core

Version: 10.25.38.1
Source: Autopsy 4.21.0

Plugin Description

This is the core Autopsy module. The module contains the cc needed for the bare application to run; the RCP platform, wir bindings, datamodel / storage, explorer, result viewers, conte framework, reporting, and core tools, such as the file search.

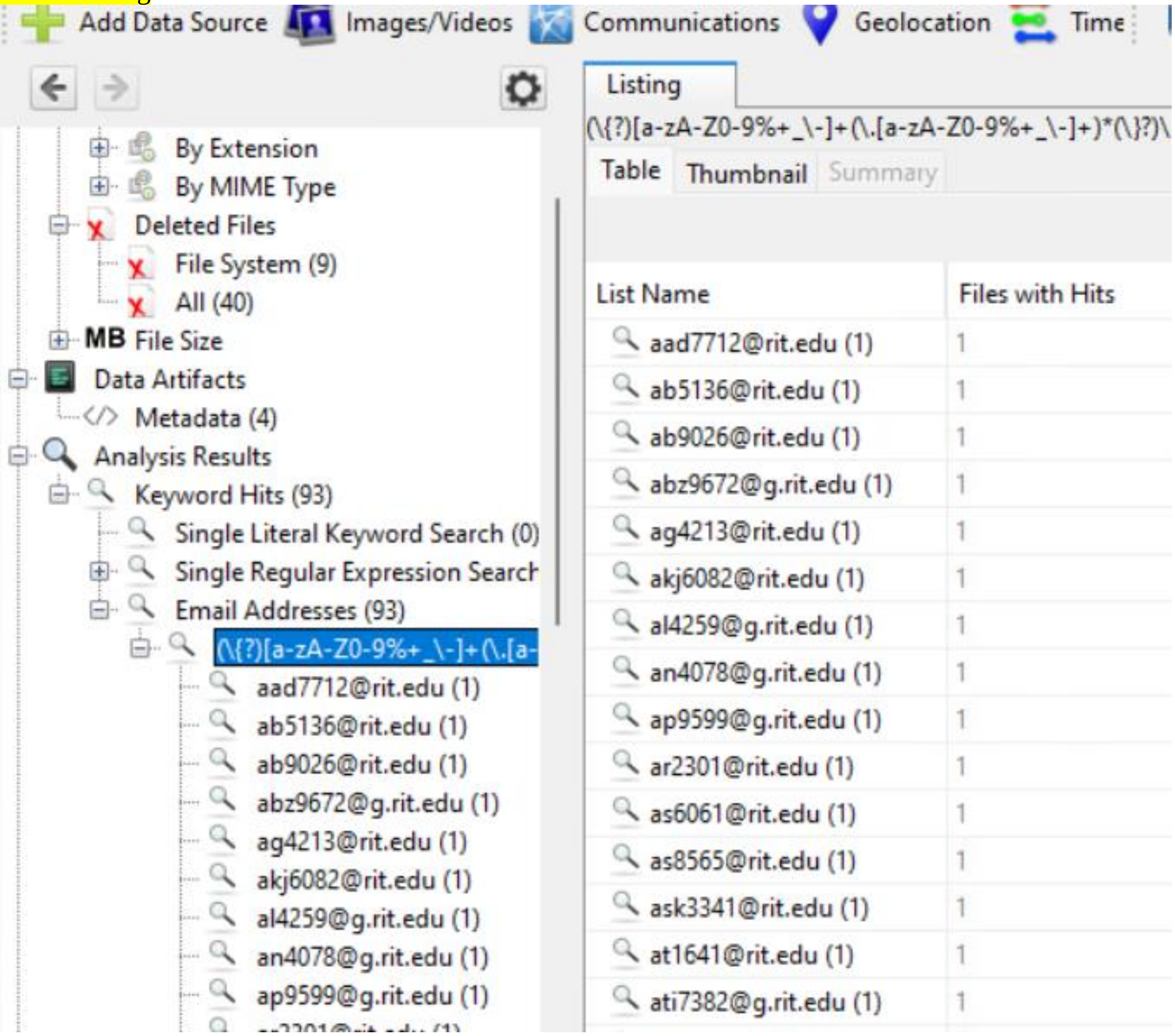
2- Image/Video Gallery



/img_Linux_Financial_Case.001/vol_vol2/$CarvedFiles/1/f0002976.xlsx/ -- 0 hash set hits / 1 files

	A	B
1		
2		**Name**
3		
4		mayuresh
5		Isaac
6		Pratith Kanagaraj
7		Yogeesh Seralathan
8		Teija Mortvedt

3- Email Listing



(\{?)[a-zA-Z0-9%+_\-]+(\.[a-zA-Z0-9%+_\-]+)*(\}?)\

List Name	Files with Hits
aad7712@rit.edu (1)	1
ab5136@rit.edu (1)	1
ab9026@rit.edu (1)	1
abz9672@g.rit.edu (1)	1
ag4213@rit.edu (1)	1
akj6082@rit.edu (1)	1
al4259@g.rit.edu (1)	1
an4078@g.rit.edu (1)	1
ap9599@g.rit.edu (1)	1
ar2301@rit.edu (1)	1
as6061@rit.edu (1)	1
as8565@rit.edu (1)	1
ask3341@rit.edu (1)	1
at1641@rit.edu (1)	1
ati7382@g.rit.edu (1)	1

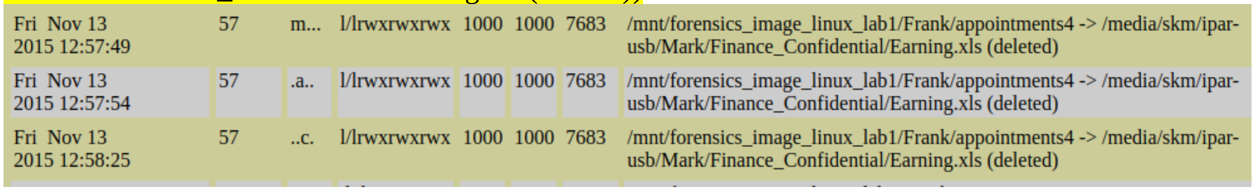
2) Compare the Windows Autopsy with the Linux Autopsy and provide your comments with two or three sentences. (6 points)

Windows Autopsy and Linux Autopsy are both powerful forensic analysis tools with similar core functionality. Additionally, while the overall features and user interface are similar, there may be slight differences in appearance and functionality to accommodate the specific characteristics of each operating system.

Part 4. Report (14 points)

Read the case scenario again and provide a short report that includes:

1. Your statement and evidence that indicates Frank may have read Earnings.xls. (8 points)
   The symlink found in Frank's directory pointing to "Earnings.xls" in a confidential directory strongly suggests that Frank accessed or attempted to access the document. This symlink creation acts as direct evidence of Frank's interaction with the file.
   (/mnt/forensics_image_linux_lab1/Frank/appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted))



Fri Nov 13 2015 12:57:49	57	m...	l/lrwxrwxrwx	1000	1000	7683	/mnt/forensics_image_linux_lab1/Frank/appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)
Fri Nov 13 2015 12:57:54	57	.a..	l/lrwxrwxrwx	1000	1000	7683	/mnt/forensics_image_linux_lab1/Frank/appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)
Fri Nov 13 2015 12:58:25	57	..c.	l/lrwxrwxrwx	1000	1000	7683	/mnt/forensics_image_linux_lab1/Frank/appointments4 -> /media/skm/ipar-usb/Mark/Finance_Confidential/Earning.xls (deleted)

2. Why is Frank able to read a confidential document? (3 points)
   Frank's ability to create a symlink to a confidential document suggests he had read or execute permissions for the directory where "Earnings.xls" was stored, possibly due to misconfigured directory permissions or improper access control setup.
3. How do you change the permissions, so that the “Earning.xls” file will not be accessible by others? (3 points)
   Enhancing the security for "Earnings.xls" involves strict permissions management for both the file and its parent directory. By setting restrictive access controls, you ensure that only the file's owner has the necessary permissions to interact with it, effectively minimizing unauthorized access risks. This strategy extends to the folder containing "Earnings.xls," providing an additional layer of protection for sensitive information.

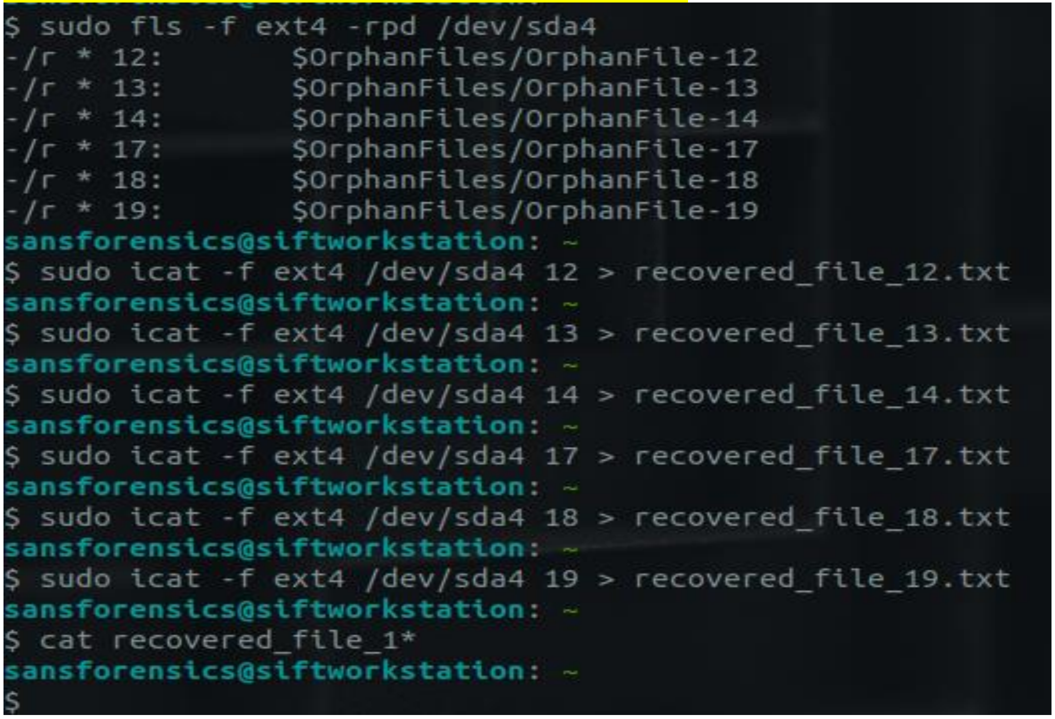
Part 5. Bonus (20 points)

Analyze deleted files in the ext4 filesystem.

1. Create a small ext4 partition on SIFT VM, create some files, a directory, and a couple of files in the directory, and delete some files.

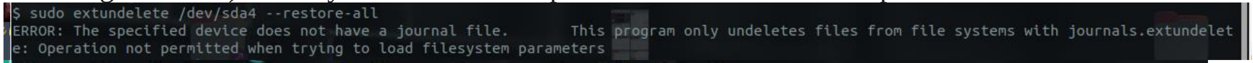
2. Use Sluethkit commands *fls* to list the inode of the deleted file. Show the inode content. Are you able to recover the content of the deleted files? Explain

I couldn't recover the data. When a file's deleted, its recovery depends on whether the system hasn't yet written new data over its spot. In my case, it seems that space was already used up, making recovery not possible.

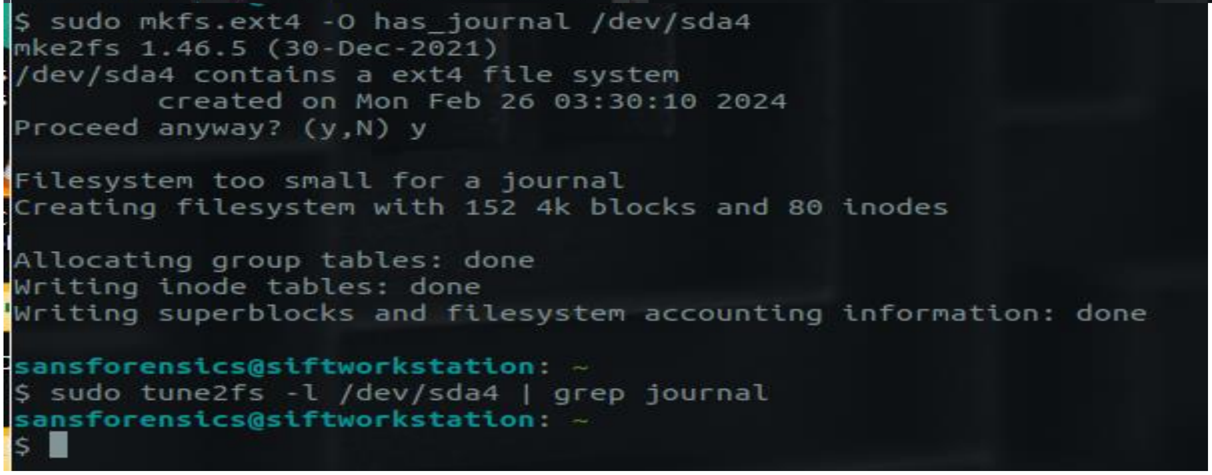


```
$ sudo fls -f ext4 -rpd /dev/sda4
-/r * 12:       $OrphanFiles/OrphanFile-12
-/r * 13:       $OrphanFiles/OrphanFile-13
-/r * 14:       $OrphanFiles/OrphanFile-14
-/r * 17:       $OrphanFiles/OrphanFile-17
-/r * 18:       $OrphanFiles/OrphanFile-18
-/r * 19:       $OrphanFiles/OrphanFile-19
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 12 > recovered_file_12.txt
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 13 > recovered_file_13.txt
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 14 > recovered_file_14.txt
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 17 > recovered_file_17.txt
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 18 > recovered_file_18.txt
sansforensics@siftworkstation: ~
$ sudo icat -f ext4 /dev/sda4 19 > recovered_file_19.txt
sansforensics@siftworkstation: ~
$ cat recovered_file_1*
sansforensics@siftworkstation: ~
$
```

3. Use *extundelete* to try to recover the deleted content (Note: undelete using *extundelete* is not guaranteed). Show your results and explain how *extundelete* attempts to undelete the content.



```
$ sudo extundelete /dev/sda4 --restore-all
ERROR: The specified device does not have a journal file.       This program only undeletes files from file systems with journals.extundelet
e: Operation not permitted when trying to load filesystem parameters
```



```
$ sudo mkfs.ext4 -O has_journal /dev/sda4
mke2fs 1.46.5 (30-Dec-2021)
/dev/sda4 contains a ext4 file system
        created on Mon Feb 26 03:30:10 2024
Proceed anyway? (y,N) y

Filesystem too small for a journal
Creating filesystem with 152 4k blocks and 80 inodes

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

sansforensics@siftworkstation: ~
$ sudo tune2fs -l /dev/sda4 | grep journal
sansforensics@siftworkstation: ~
$ 
```

So the *extundelete* didn't work with me because of the journal I tried to install it using this command $ **sudo mkfs.ext4 -O has_journal /dev/sda4,** but it didn't work with me due the fact that I tried to check if the partition has the journal using GREP but again nothing worked. Also, I tried to change the size by resizing the parititon but didn't work too I used this command **(sudo resize2fs /dev/sda4 1G)**