Design and Development of CLI for SleuthKit: A Cyber Forensics Framework

Dilpreet Singh Bajwa*, Gurpal Singh Chhabra**

* Department, Institute Name
** Department, Institute Name, if any

Abstract- There are many cyber forensic tools available for extraction, making copy of original media and for analysis. Tools are inherent part of any cyber forensic investigation and they must be based on proven methodology and techniques admissible under legal procedure.

The Sleuth Kit (TSK) is a popular open source cyber forensic tool constitutes a library and collection of command line tools that allow user to investigate disk images. These command line tools are difficult to use and the user have to use each one only independently. Output is also not saved for future reference and analysis.

In this work, a common command line user interface is created for command line tools of sleuth kit and automates the process. Some other tools for hash calculation are also incorporated in the system to make it more efficient.

Index Terms- Cyber Forensics, Digital Forensics, Computer Forensics, Cyber Forensics Tools, Sleuthkit, Open source Cyber Forensic tool.

I. Introduction

Due to integration of computer and communication technology and the fast development of digital technology have made significant changes to computer world. Firstly, the effectively and efficiently processing capability of computer made it most important tool for data processing. With the new technology storage capacity is also increased day by day. As a result, more and more data is processed and stored in computer systems. Secondly, the internet influence is so much in our daily life from simple email communication to banking transactions, online shopping, surfing, social networking etc. With this increase in computer and internet usage, the crime related to computers is also increased gradually. To counter these criminal activities and also to prove crime in court of law, a new field: "Cyber Forensics" came in to existence which incorporates procedure, tools and techniques to find the evidence against cyber criminals and prove it in court of law.

Cyber forensics is defined as "The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations [1]".

For cyber forensic investigation cyber forensic tools are now used on a daily basis by examiners and analysts within local, state and National law enforcement agencies. Right kind of tool is in your hand to create a best sculpture, similarly right kind of tools are required to properly investigate the case. The authenticity and reliability of the gathered evidences rely to great extent on the used forensic tools. If used tools are not reliable, the result produced by these tools will not be considered reliable [2].

In the present work, a CLI for sleuthkit which is a open source cyber forensic tool is created and automate the process. Some other tools for hash calculation are also incorporated in the system to make it more efficient. Guidance and help is provided while we are executing these command line tools further output just not shown on screen, in addition it saves in a file for further analysis. Secondly whole session and steps followed by user as cyber forensic expert is also saved. Figure: 1.1 given below helps to understand the idea.

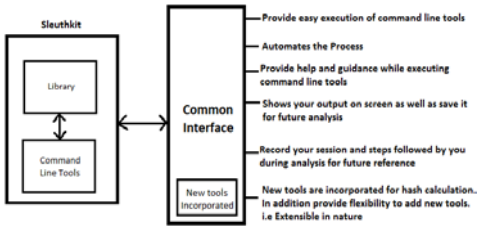


Figure: 1.1: Higher Level Model of CUI for Sleuthkit

So, the interface created provides easy execution of command line tools, automate the process provide flexibility to add new tools other than TSK.

This paper is organized in to seven sections: First section provides introduction and background information, second section discuss, is it fine to use open source tool instead of commercial tools. In Third section problem statement is given. Fourth section provides proposed Design. Fifth section discusses implementation, Sixth section show working and experimental results and in Seventh section conclusion and future scope is given.

II. Open Source Vs Proprietary Tools

In cyber forensics investigation Interpretation, analysis and documentation are important steps to find the evidence and to prove it in court of law. For analysis of digital devices several commercial and open source forensic tools are available. Commercial tools are software products provided by different organizations. Law enforcement agencies are using these tools for cyber forensics. Education institutes which running course in

cyber forensics also use these proprietary tools. These commercial tools are very expensive and can be purchased on license for particular duration. After expiry you again have to renew the license. They are easy to use and proper documentation and support is available for these commercial tools usage. On the other hand open source cyber forensic tools are also available which also equally efficient [3] but the disadvantage of using an open source tools is lack of support and proper documentation. It is also finding difficult to use these tools. First, most Open Source Solutions (OSS) framework were adopted from system utilities such as disk backup, file system detection and system check that were not designed for computer forensics usage. Second. Investigator must have high-level information about computer architecture which posed serious risk for novice or intermediate level investigator. Third, no GUI and, output provided by open source tools is text based and usually very hard to understand. Fourth in many cases, an expert has to use more than one tool to obtain evidence [4].

But there are several benefits attached in using open source cyber forensic tools: You don't have to purchase them, No license fee is required, and you can use them anywhere i.e. outside the licensed lab, for students, educationist and the organizations who don't afford commercial tools, the open source tools are the best choice. In [4], author states that despite the availability of commercial computer forensics software, most computer forensics investigators prefer to use the freely available open source solutions (OSS) due to various reasons: OSS freely available, OSS can be used to perform preliminary acquisition Information gathering while most commercials can only do post acquisition information gathering. Further the code is available for these open source forensic tools, you can optimize and modify them according to your requirement, you can also understand the internal working and code and how results are produced which is not possible with commercial tools. You authenticate them in court of law. Most of these Open Source tools in no way are less reliable and effective, when compared with the proprietary suits [3]. Open source software continues to be one of the most widely used tools in computer forensics [5].

In the scenario to learn, to experiment, to understand internal working and code, open source cyber forensic tools are the best bet. Open source tools with proper combination also provide all features provide by any commercial tool.

III. Problem Statement

As it is clear from above discussion that open source tools provide a good alternative for computer forensics. The Sleuthkit (TSK) is open source cyber forensic software and constitutes a C library and collection of command line tools for file and volume system forensic analysis [6]
Following are the main Command Line Tools available in Sleuthkit [6]:

- **fsstat:** Shows file system details and statistics including layout, sizes, and labels.
- **ffind:** Finds allocated and unallocated file names that point to a given meta data structure.
- **fls:** Lists allocated and deleted file names in a directory.
- **icat:** Extracts the data units of a file, which is specified by its meta data address (instead of the file name).
- **ifind:** Finds the meta data structure that has a given file name pointing to it or the meta data structure that points to a given data unit.
- **ils:** Lists the meta data structures and their contents in a pipe delimited format.
- **istat:** Displays the statistics and details about a given meta data structure in an easy to read format.
- **blkcat:** Extracts the contents of a given data unit.
- **blkls:** Lists the details about data units and can extract the unallocated space of the file system.
- **blkstat:** Displays the statistics about a given data unit in an easy to read format.
- **blkcalc**: Calculates where data in the unallocated space image (from blkls) exists in the original image. This is used when evidence is found in unallocated space.
- **jcat**: Display the contents of a specific journal block.
- **jls**: List the entries in the file system journal.
- **mmls**: Displays the layout of a disk, including the unallocated spaces.
- **mmstat**: Display details about a volume system (typically only the type).
- **mmcat**: Extracts the contents of a specific volume to STDOUT.
- **img_stat:** tool will show the details of the image format
- **img_cat:** This tool will show the raw contents of an image file.
- **hfind:** Uses a binary sort algorithm to lookup hashes in the NIST NSRL, Hashkeeper, and custom hash databases created by md5sum.
- **mactime:** Takes input from the fls and ils tools to create a timeline of file activity.
- **sorter:** Sorts files based on their file type and performs extension checking and hash database lookups.
- **sigfind:** Searches for a binary value at a given offset. Useful for recovering lost data structures.

These tools perform some specific function and we can execute them independently only. We can also use many arguments with these command line tools which make their function more specific. So sleuthkit contains rich set of these command line tools which are used for disk analysis.

The file system tools are used to examine file systems of a suspect computer. It works on both Windows and UNIX platform. The volume system (media management) tools are used to examine the layout of disks and other media. With these tools, you can identify where partitions are located and extract them so that they can be analyzed with file system analysis tools. When performing a complete analysis of a system, we all know that command line tools can become tedious to use and you must know how to use them [6]. The basic idea behind working of sleuthkit command line tools is given below in Figure 3.1:

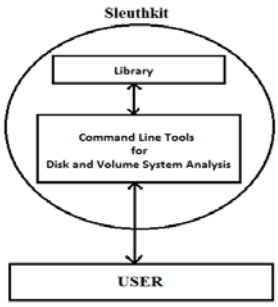


Figure: 3.1: User Interaction Scenario with Sleuthkit

The user can execute command line tools directly and perform disk and volume system analysis.

We observed following problems while using these command line tools:

- Each one can only be executed independently.
- Lack of help and support while executing these command line tools.
- Output is shown on screen only, not saved for further analysis.
- No case management is available or facility for session record while executing these command line tools.
- Not a common interface available for all command line tools to show these are the tools available to use and you can select one of them according to your requirement at particular time.
- We can only use command line tools available with TSK.

The problem is that it is difficult to use these command line tools directly without any prior knowledge. Secondly you have to execute each command line tool independently and also the output is shown on terminal only, further what steps you are performing in sequence is not recorded or saved for future reference and analysis. Here to optimize this direct interaction of user with sleuthkit, a command line user interface is created to access these command line tools in easy way and automate the process. The output is saved for further analysis and whole session is also recorded for future reference. You may also incorporate new tools also other than TSK in the framework.

IV. Proposed Design

After considering the points mentioned in problem statement, we proposed and develop a common CLI which overcome all above limitations and provide us following benefits:

- Commands can be executed in collaboration.
- While you are executing command, at each step you get the help.
- Output is not only shown on screen but it can also save for later analysis.
- Case Management is available, you can record your whole session i.e what commands and what steps you follow while doing analysis.
- Common interface is provided for command line tools.
- We can also incorporate other tools in to the system and use them.

According to proposed solution, a framework is designed that is used to access sleuthkit command line tools through a common interface and also add more tools which are not part of sleuthkit. This framework is useful in sense as it is simple provides help to user at each step while executing the tools and automates the process. Secondly the results are saved in different files corresponding to each tool executed by user and results can be seen later on for analysis and other purposes. One step ahead this framework provides case management that is whole session records for future reference and documentation. It records what steps followed, commands executed and their corresponding output for a complete session i.e. until you exit from the system. Figure 4.1 shows block diagram of proposed Framework.

V. Implementation

For implementation of proposed Command Line Common User Interface for Sleuthkit, Shell Programming Language of Linux is used. The reason to choose this scripting language is that firstly sleuthkit command line tools are meant to execute on Linux platform, Secondly it is easy to use and implement shell programming, it has vast variety of features and commands of Linux further powerful features like pipeline, redirection, regular expression and combination of Linux commands can also used which makes it unmatchable specifically for purpose like computer forensics. Currently implementation of img_stat, mmls, mmstat, fsstat, fls command line tools of sleuthkit is available and in future all command line tools present in sleuthkit will be implemented:

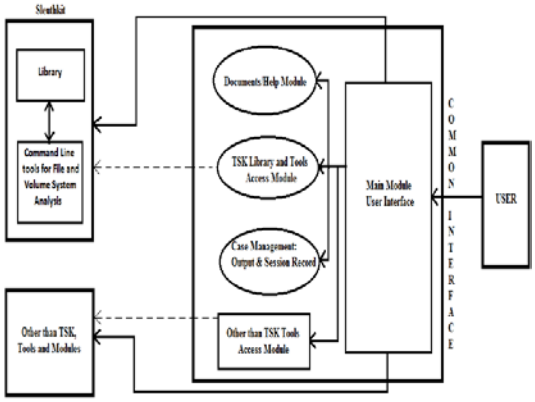


Figure 4.1: Block Diagram of proposed CLI for SleuthKit

In addition to these tools, the framework also incorporate tools used for Hash Calculation and these tools are not part of sleuthkit. We can also calculate SHA-224, SHA-256, SHA-384 and SHA-512 hash values corresponding to some file or image in addition to MD5 and SHA-1. The interface further provides you flexibility to add more tools when required.

The Interface also provides a very important functionality which is necessary from point of view of cyber forensic investigation. It provides complete case management which help in documentation of case investigation automatically and also in

future reference and analysis. For a complete session i.e. from invoking of this common interface up to exiting, it records everything from steps followed up to commands executed by user, all are saved. Separately when the user executes commands their output is also saved in different file corresponding to each command. It also makes execution of these commands easy for a beginner as well as professional by providing help and guidance at each step. You can also get complete description of tools and their various arguments before using them or while using them. So it provides a user friendly environment for users.

Broadly we divide the interface in to five modules:

First module (Main Module) is main module corresponding to this module a script file main.sh is used and all other modules are directly connected to it and it is also responsible for start and providing first interface to user.

Second module (TSK Library and Tool Access Module) is very important; it is used to provide access to user for command line tools of sleuthkit through the interface. It provides option to choose between available list of tools user want to use at particular time. Corresponding to each command line tool there is separate script file to access it. For e.g. if the user want to use img_stat tool of sleuthkit then corresponding to it script file imgstat.sh is available which works in background to help in accessing the tool and providing the options which you can use with the command. it also responsible for providing help during execution in collaboration with Document/Help Module.

Third Module (Document/Help Module) is responsible for providing documentation and complete description regarding tools available for use when required. It also provide help and description regarding various arguments/options available with command line tools while executing the tool in collaboration with corresponding script file of tool.

Fourth Module (Session Record Module) is responsible for case management. It records complete session and saved it for future reference. It also saves output of all command line tools in separate file corresponding to each tool. It works simultaneously with scripting file available for each tool.

Fifth Module (Other Than TSK Access Module) is used to provide access to user for tools (which are not part of TSK) through the interface. It provides options to choose between available lists of tools, user want to use at particular time. Corresponding to each command line tool there is separate script file to access it. It provides flexibility to add new tools and access to them. This module is expandable and user can incorporate new tools when required.

So it is clear from our module description that how modules in interface work together and specifically what they do. Figure 4.2 given below shows interaction and integration of these modules:

Flowchart Working is described as: Now in this part, it explains that how control of framework flows from start to end with the help of flowchart diagrams shown in Figure: 5.2 to 5.6 which is helpful in getting to know about the flow of control and working of interface in depth.

In actual the flowchart from Figure 5.2 to 5.6 is complete one flow chart for whole interface. So all are connected with each other at different point and can be consider as a single flowchart while observing flow of interface.

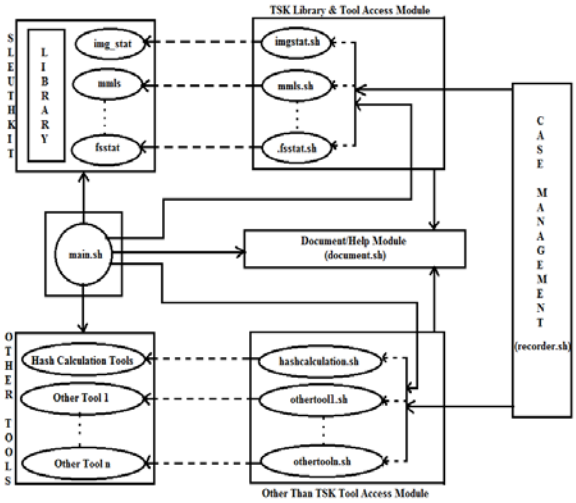


Figure 5.1: Interaction and Integration of Main Script Modules

Flowchart is divided in to four parts, First part shown in Figure 5.2 shows flow of main module and what options it provide at start. It has main three options. Option 1 is choose to get Help and Description about all Sleuthkit command line tools. 2nd option is the main option chosen by user to see list of all available tools under this frame work and from there user choose to execute them. 3rd option is used to exit from the interface. Whenever user chooses to exit during session at any stage, the whole session is saved for future analysis and reference.

Second part demonstrated in Figure 5.3 and Figure 5.4. Third part is demonstrated in Figure 5.5 and Figure 5.6 which shows the most important part of our frame work i.e. selection of tool of user's choice and its execution. When user selects option 2 from main page interface (Figure: 5.2) i.e. choose to use the tools then the system follows the control mention in the flowchart shown in Figure 5.3. Before moving forward it asks you for case name. Whole session now records under this case name. After providing case name, the interface shows all the tools which are available for use, user can select one if want to run otherwise also select exit. When user chooses any tool then options corresponding to selected tool are shown. Corresponding to each command line tool of sleutkit these options are shown: if you want to run the tool then select 1, 2nd option is also important here as it provides user to build its own command line and directly run it on command prompt, if user select 3rd option then get complete description about the tool and various arguments which can be use with this tool to make it more specific,4th option take one step back to see again list of available tools and from there you again continue to choose same or any other tool, 5th option takes you to home screen and 6th takes to exit point.

If user select to run any command line tool then first it asks about image full path (Figure: 5.5) i.e. the image on which user wants to apply forensic investigation and also arguments are shown which can be used with the tool and the interface also asking whether want to use one of these argument or not. If user select no than in that case tool is run with default arguments otherwise if user select yes than interface asks for the argument user want to use and prepare command line according to that and execute the tool. When user executes some command, the output

is shown on screen and also message is displayed that your output is also saved and storage location is also specified.

If user select option "Tools other than TSK" in Figure 5.3, it shows all available tools which are not part of sleuthkit but incorporated in to framework to provide more functionality during cyber forensic investigation. Now your control goes to Figure 5.6, Currently Hash calculation tools are incorporated in to framework so they are shown with corresponding options. Either user choose to use tools for hash calculation or move back to see list of all tools or select 3rd option to go back to home or choose 4th option to exit.

If in Figure 5.6 if user selects option 1 and chooses to run hash calculation tools then interface shows all available options of hash calculation from md5 to sha-512. These tools are part of Ubuntu, like this we can incorporate other tools in to our interface. User can either select one of the hash calculation tool or choose option 6 to apply all hash calculation commands and calculate all hash values but before executing one or all tools system prompts user for name of file or image for which user wants to calculate hash. User can also select option 7 which takes user back to see all available tools again or user can also choose 8th and 9th option for moving back to home or to exit.

In any case from start to end if user can't pick the right option then error handling mechanism display message wrong option chosen and again prompt to same screen so that user choose the right option.

Refer to Figure 5.5, if we choose to calculate hash than interface ask for file or image name on which user want to perform hash calculation and after giving image or file name, it gives you output on screen and also saves it, further message is also displayed that your output is saved and where it saved.

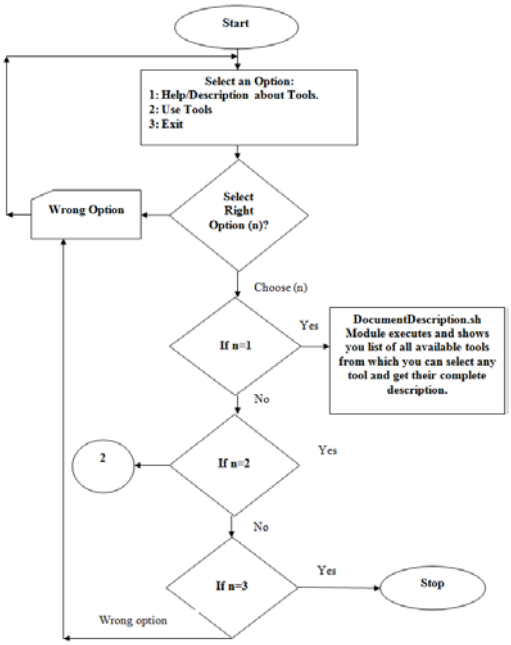
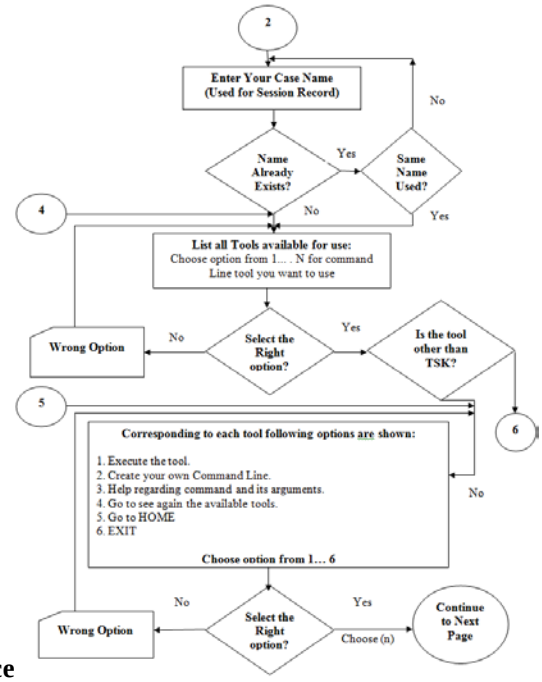


Figure: 5.2: Flowchart for Main



Interface

Figure 5.3: Flowchart of Interface for tools available for Use and Command Execution-1

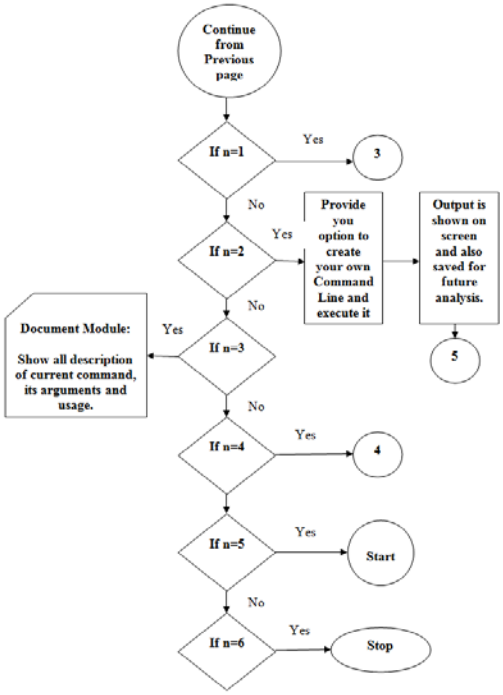


Figure 5.4: Flowchart of Interface for tools available for Use and Command Execution-2

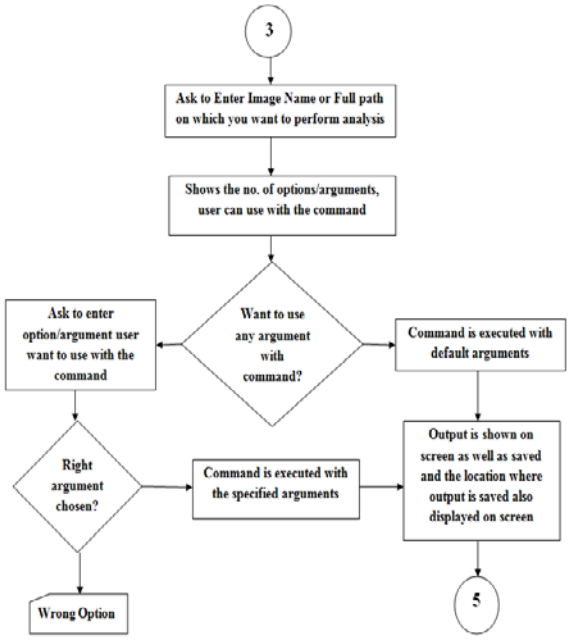


Figure 5.5: Flowchart for execution of Tools.

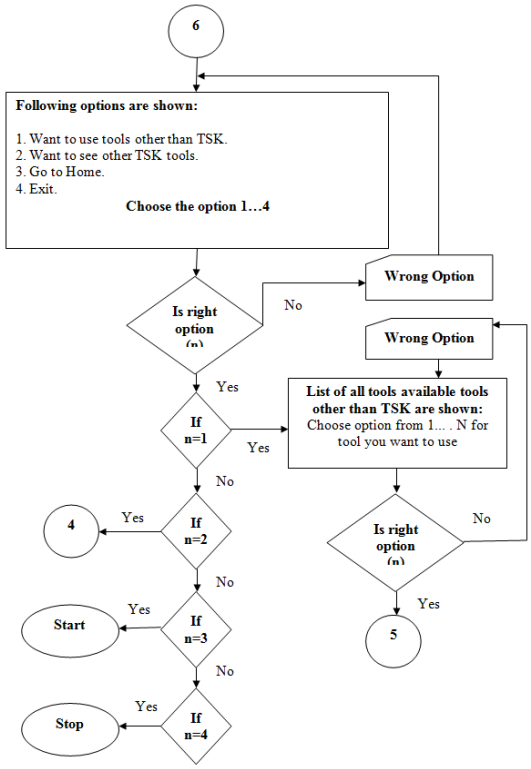


Figure 5.6: Flowchart for Interface of Tools Other Than TSK available for Use and their execution.

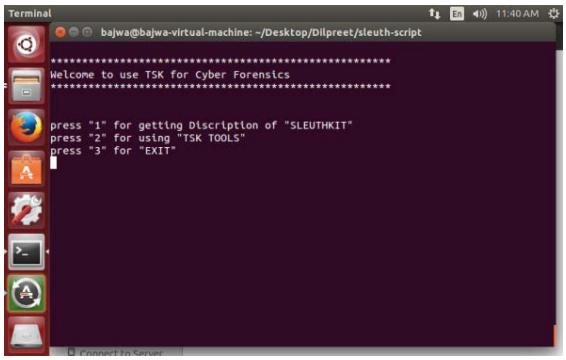
VI. Experimental and Test Results

After coding and integration of Common CLI for sleuthkit is used and it is observed that interface is working properly corresponding to the command line tools which are implemented. Results are shown and also saved, commands are executed successfully, and session is recorded. All the options shown in interface are working correctly and flow of control is also following the valid path.

Here in this section screenshots are shown while using the interface. As shown in screen shots interface is working quite well. It provides all features and overcome all limitations which described earlier. It is also observed that interface has potential to incorporate more features and functionality which we will definitely incorporate in near future. Not all screen shots included corresponding to all tools. Only limited screen shots are included.

Screen Shots:

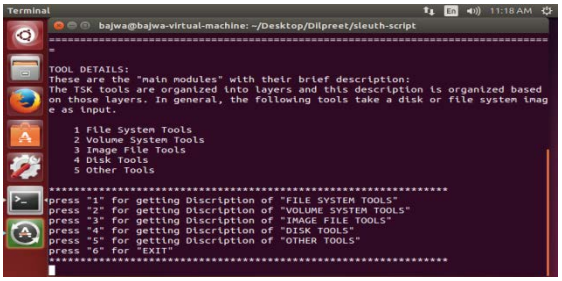
Screen Shot 6.1: HOME



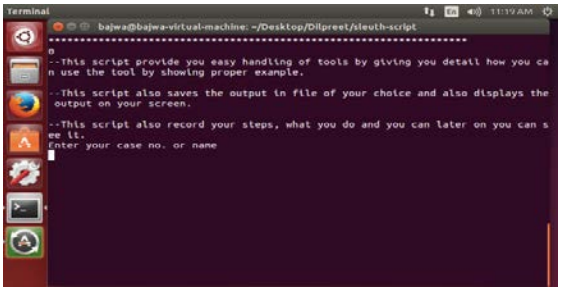
Above screen shot is first main home page, Here it provides three options regarding getting description of sleuthkit, want to use tools or want to exit, User have to choose only one option by pressing corresponding number. After that it takes you to next level.

Screen Shot 6.2: Description/Help for Tools

After choosing option 1 from home page, user move to this screen which provides link for complete description of sleuthkit command line tools category wise. To move forward and to get complete description of tools, users again have to select option from the given ones.

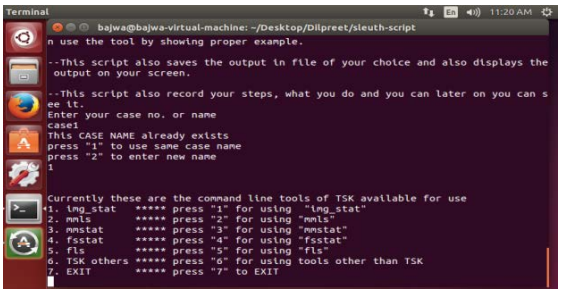


Screen Shot 6.3: Asking for Case Name for Session Record



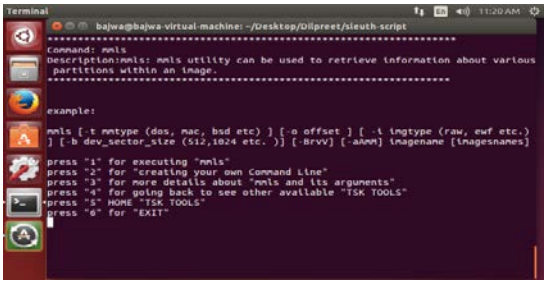
When user selects option 2 i.e. choose to use the tools then it moves to this screen. Before moving forward system asks for case name. Your whole session now records under this case name.

Screen Shot 6.4: Shown Tools available to execute



This screen shows all the tools which are available for use, user can select one if want to run otherwise also select exit. Currently img_stat, mmls, msstat, fsstat and fls are implemented, so they are shown. At 6th option there is choice to select tools other than TSK.

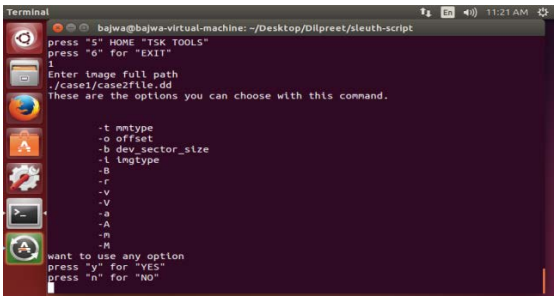
Screen Shot 6.5: Asking for execution of Command Line tool mmls and also shown how it will be used.



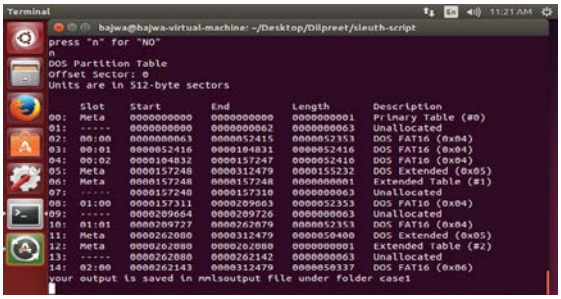
From above screen if we select one tool then this type of screen appears. Here we choose the mmls tool so options corresponding to mmls are shown. Corresponding to each command line tool of sleutkit options shown above is displayed on screen.

Screen Shot 6.6: Asking for image full path on which command is applied, also shown the no. of options you can use with the command and asking for whether you want to use any option.

If user selects to run any command line tool then screen similar to this appears. In previous screen 6.5, if we choose to run mmls tool so this screen appears corresponding to mmls tool. First it asks about image full path i.e. the image on which user wants to apply forensic investigation and also arguments are shown which can be used with the tool. Here arguments for mmls tools are shown and also asking to user whether want to use one of these argument or not. If user select no than in that case tool is run with default arguments otherwise if user select yes than interface asks for the argument user want to use and prepare command line according to that and execute the tool.



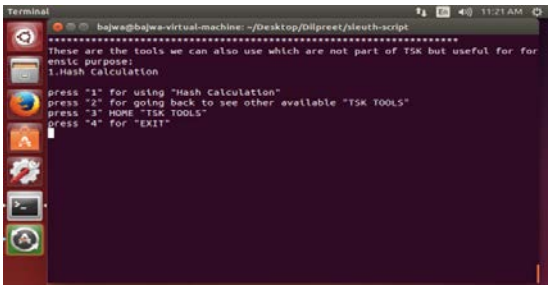
Screen shot 6.7: Output is shown corresponding to execution of command mmls and also told that where your output is saved.



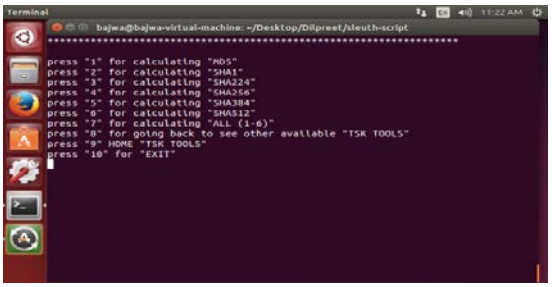
This type of output screen is shown corresponding to result obtained after execution of any tool and also message is displayed that your output is also saved and storage location is also specified. Above screen shows the output corresponding to execution of mmls tool.

Screen Shot 6.8: This screen shows the tools available within this frame work other than TSK and also provides various options you can choose.

Screen below appears when user selects to use tools other than TSK. It shows all available tools which are not part of sleuthkit but incorporated in to framework to provide more functionality during cyber forensic investigation.

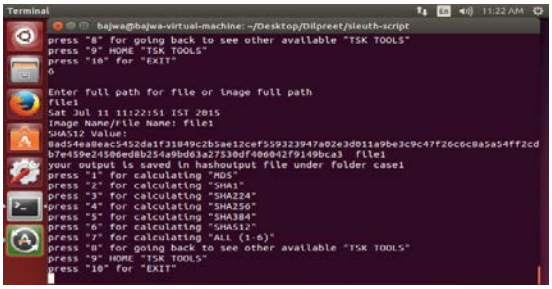


Screen Shot 6.9: Shows Hash Calculation tools available within this framework.



If from previous screen 6.8, user selects option 1 and chooses to run hash calculation tools then following screen appears. It shows all available options of hash calculation from md5 to sha-512 but before executing one or all tools system prompts user for name of file or image for which user wants to calculate hash. User can also select option 7 which takes user to screen 6.4 to see all available tools or user can also choose 8th and 9th option for moving back to home or to exit.

Screen Shot 6.10: Asking for File or image name for which hash will be calculated, Result is also shown and also shown that where your output is saved.



From previous screen 6.9, if user choose to calculate hash than this screen appears asking for file or image name on which user want to perform hash calculation and after giving image or file name, it gives you output on screen and also saves it, further message is also displayed that your output is saved and where it saved. Currently from previous screen we choose to calculate sha-512 so corresponding output is shown.

So from above screen shots it is clear that our common CLI for sleuthkit is working properly and efficiently. In addition to this the framework also check errors, if any wrong argument is passed or any wrong option is chosen by user and corresponding message is displayed. Second, the whole session from start to end (until user chooses to exit) is recorded under the case name provided by user.

Above small set of screen shots are shown, for each tool there are around 5-6 main screen shots are available, so it is not possible to include all screen shots but from above screen shots you get an idea how actually this interface is working.

VII. Conclusion

This work is focused on open source forensic tool because they are free and their code is also available which can be studied, modified and expanded according to the requirement. Sleuthkit is chosen because it is very popular tool for disk and volume system analysis in forensic investigation. It constitutes set of command line tools. As mentioned earlier these tools are not so user friendly, it is difficult to use them without any prior knowledge. Therefore a CLI interface is developed for users who want to use these command line tools easily. Actually process is automated and at each step options are provided to users and user can easily select one of them. At the backend, the interface can handle the command line tools for user. When these tools are used through this interface, at each step user get help about how to use these tools and what arguments can be use with the

command. CLI also saved the output and record the session for future analysis. This interface also provides flexibility to incorporate and use tools other than TSK under the same common framework which further enhance the functionality of investigation process.

Future Scope: Some tools of sleuthkit are not implemented in this interface, in future work, trying to implement all remaining tools of sleuthkit and also add more tools other than TSK to enhance the functionality of framework as well as cyber forensic investigation. Currently it is tested on Ubuntu , we try to create interface for windows on same pattern and there is also possibility of showing the output in graphical form.

REFERENCES

[1] Gary Palmer et al. ,"A Road Map for Digital Forensic Research", Report From the First Digital Forensic Research Workshop (DFRWS),Utica, New York, August 7-8, 2001.

[2] Khidir M. Ali, "Digital Forensics Best Practices and Managerial Implications" Presented at Fourth International Conference on Computational Intelligence, Communication Systems and Networks, 2012, IEEE.

[3] Manuel Delgado, Manuela Aparicio, Carlos Costa, "Using Open Source for Forensic Purposes",Proceedings of the Workshop on Open Source and Design of Communication, PP 31-37, ACM, New York,USA-2012.

[4] Azril Azam, Raja Mariam Ruzila, "Preliminary Acquisition Information Gathering on Computer Data Storage: Open Source Software (OSS) vs. FIRST DiskImager", International Symposium on Information Technology, 2008. (Volume:1), Kualalumpur Aug-2008.

[5] Byfield, Bruce, "The two-edged sword: Legal computer forensics and open source", 11April. 2005. News Forge. 13 Feb. 2006 http://software.newsforge.com/software/05/04/052052235.shtml.

[6] Sleuthkit: http://www.sleuthkit.org/sleuthkit/docs.php

AUTHORS

First Author – Dilpreet Singh Bajwa, DCA Deptt, CGC, Punjab (India), dilpreetbajwa10@gmail.com
Second Author – Gurpal Singh Chhabra, CSE Deptt, Thapar University, Punjab (India), gurpal.singh@thapar.edu