Real Time Activity Logger: a User Activity Detection System

Hitesh Kumar Sharma, Ishu Khanchi, Nihal Agarwal, Pawas Seth, Prashant Ahalawat

Abstract—With the size of the enterprise network is growing, more and more computer terminals, internal information leakage and security threats emerging security incidents and daily management to bring a lot of pressure. Business users need a tracking system to address these security issues. A technique and framework for checking and following the exercises of a client of a PC. Affiliations are set up between the conditions of certain PC framework parameters and explicit exercises. The present movement of the client is then definite by the framework dependent on the present condition of the PC framework. As the condition of the PC framework changes, changes in the client’s action are checked and followed. The exercises are followed related to time in order to record the time spent on every movement. Through this project we are developing the understanding of the working and the basics of the Linux operating system. We are analysing the need of monitoring the system and solving the issues related to security, unauthorised access and other threats. Through this tool we are trying to address these issues.

Key Words— Tracing, Monitoring, Log files, Database, Server

I. INTRODUCTION

These days security is to be given highest priority and that too without compromising user's privacy. But still, this is a matter of concern for a lot of users and administrators. Many software's/installers came into market but each one has its own issues and disadvantages. Many suspicious user activities can compromise security across the network and will lead to data breaches, so finding the root cause of the breach or any other problem regarding file system. Manual Analysis increases the time to solve the threats. Keeping this in mind the ultimate need of a tracer to track down the activities of any third party given access by the user/administrator, this project is a utility tool. This utility tool is made to monitor and trace the activities done on the user’s system by any third party and keeping the logs of those activities in the log files. We have used the concept of different Linux commands, and have implemented their functionality in C programming language. We have integrated all the functionality to make the application user-friendly and user and administrator can use for different purposes like knowing the uptime of the system or for security concerns and many more. This is also used for monitoring the activities of the user.

II. PROBLEM STATEMENT

The question why this tool is needed can be answered in terms of security. Whenever any third party will have access to administrator’s system, he can keep a check on it so as to assure every possible security. This tool will be very beneficial as it traces user’s activity, monitors user’s usage, maintains log files uses databases and stores the files on server so that administrator can access them from anywhere and this also leads to less memory usage as data will be stored on the server. Data breach can be avoided. Non-centralized and manual log analysis increases the time it takes to remediate threats. Finding the root cause of an incident, and tracing it to a suspected user, prevent downtime of the businesses.

III. LITERATURE REVIEW

Activity fetcher like any other activity tracer, is used for monitoring the activities of the user working on Ubuntu (LINUX OS). Created with a main purpose of security and data privacy, this tool will be useful for the users working with various businesses. The main objectives are:

- Tracing down users’ activities on the system.
- Monitoring activities done by the user on the network.
- Maintaining and accessing the log files that store the session time.

Similarly several other activities like web based monitoring, screen monitoring, kernel monitoring etc. can be included in this tool. This tool saves memory as data is stored in databases on the server. Data can be easily accessed by the administrator and can be monitored from wherever on the server.
Focusing on various objectives, this tool is necessary from various point of views. Many companies will be benefitted from this tool.

IV. IMPLEMENTATION

After legitimate usage of the accessible resources, we have dissected that what are the conceivable routes through which we can discover the information of our system. On account of which we have built up our very own approach to keep up every one of the records of the opened or running application and the introduced applications inside the system by utilizing Linux commands and implanting them using C. A few commands are recorded underneath through which we can get extra data about the system:-

- uptime
- ctime
- ls
- etc.

We further continue where we had made connectivity of our code with MYSQL database by utilizing PHP. From that point onward, we are getting the information in regards to introduced and running applications and store the particular information inside the database as a table for quick recovery at the season of following the movement of the system. This code will keep running at the back-end after an explicit measure of time. The present time we have brought from the system itself and showed onto the system.

Fig. 1: (DATA FLOW DIAGRAM)

Fig. 2 Java Code Screenshot

Fig. 3: User Interface with the various options for the user to select.
Finally we have made a PHP page which works like a web portal through which we can access the database as per our motive of using the application. The databases can be uploaded on the server. Through this administrator can access databases from anywhere remotely so as to keep track of the user’s activities on the system. As shown below:
V. CONCLUSION

Our project is a model which address this issue in a comprehensive way. It would prove very beneficial in the real world as the project can work in dynamic conditions. It provides a very optimized solution to this problem very focused on the goal of monitoring the system and user activities and accessing data remotely from the server. There are various techniques came to our knowledge regarding the monitoring of the system. We can use our approach in integration with various others approaches discussed. It will finally come to the users need. And according to the users need our developed approaches can be modified.

REFERENCES