OCR Enabled Monitoring and Administration of Computers Over LAN

Rushabh Bid
Department of Information Technology
K. J. Somaiya College of Engineering
Mumbai, India

Yash Kandalam
Department of Information Technology
K. J. Somaiya College of Engineering
Mumbai, India

Pratyusha Reddy
Department of Information Technology
K. J. Somaiya College of Engineering
Mumbai, India

Yash Jain
Department of Information Technology
K. J. Somaiya College of Engineering
Mumbai, India

Ravindra Divekar
Department of Information Technology
K. J. Somaiya College of Engineering
Mumbai, India

ABSTRACT

Network surveillance has become a necessity with the increase in cybercrime. Monitoring the activities of the devices on a network is paramount to the security of an organization. It is difficult for a novice user to understand the root levels of LAN monitoring through packets sniffing. In order to remove this complexity, a simple UI driven application which will enable the administrator to observe, manage and analyze the activities of the client machines is proposed. While preventing cybercrime the system also aims to create a discipline in the activities of the users working in the organization.

Keywords

LAN, monitoring, networks, OCR, administration

1. INTRODUCTION

There is a plethora of monitoring software applications that work on the concept of screenshot sharing. In this paper, an approach of blending the administration and monitoring of devices on a local network with a clean user interface is discussed. Further, the idea to generate a score of the client machine based on screenshots using optical character recognition (OCR) is proposed to reduce the need for human attention and interruption.

The primary requirement of the system is the connection of each client to the admin. The administrator will be able to remotely monitor the applications running on client computers and also terminate them if needed. The administrator will be able to send personal and broadcast messages and shut down machines as well [3].

2. PROPOSED SYSTEM

The system proposed in this paper requires a physical connection between clients and the administrator, making it suitable for Local Area Networks. On start-up, the client machines connect to the administrator machine. For every connected machine, the administrator machine requests screenshots at a defined interval of time.

The administrator also sets keywords for sessions, and they are sent to the appropriate clients. Based on the running applications, and the comparison of the text in the screenshots with the set keywords, a relevance score is generated for the client’s work over the session. The UI is responsive to this score and is color-coded according to it.

On noticing ill behaviour, the administrator can message the client to alert the client. The administrator can view the applications running on the client and depending on the severity, stop specific applications and even shut the machine if required [3]. To communicate general information, the administrator can broadcast a message to all the clients. If there is a need to ignore a client machine for security reasons, the administrator can disconnect it after which the screenshots of that client machine will no longer be stored or sent. Lastly, on session end, the administrator can check for connected clients and shut all the connected machines to securely close the session.

3. FEATURES OF THE SYSTEM

The application is lightweight and easy to install. The use of multiple threads makes the processing faster.

Following are the features:

- Remotely monitor the activities of client machines through the screenshots received
- Generation of the score for client machines
- Demand a live screenshot of the client machine at any instance
- Send a message to any connected client machine
- Broadcast message to the connected client machines
- Remotely shutdown any connected client machine
- Disconnect a particular client machine
- View and manage processes running on the client machine

4. IMPLEMENTATION TOOLS

4.1 Tesseract - OCR

Tesseract is an open source OCR engine which was released under Apache license and sponsored by Google. It is a powerful tool which can be easily used with a variety of programming languages and also has strong community support.

Optical Character Recognition (OCR) is the technology that is used for recognition of text in an image. The following pre-processing is proposed to get the optimum efficiency from the Tesseract OCR. The computer first converts the screenshot image into a gray scale image. The image undergoes appropriate pre-processing to remove any noise present in the image. After the image is pre-processed the text in the image gets photo scanned character by character. This data is then analyzed and translated to ASCII character code. The text data that is produced has a lot of junk symbols which is cleaned using regular expressions. The final text data contains the text recognized by the Tesseract OCR which is free of junk values and is stored in a list format.

Getting clean text data using OCR is the first step of generating a relevancy score for the client machine.

4.2 Flask - Web Framework

The proposed system is developed as a web application. This results in a multiplatform, highly flexible, readily deployable and intuitive application. The web application is built to work without the internet which decreases the number of requirements to install and use the software. On running the application, a web browser opens up, with a predefined address, comprising the UI. An HTTP request will be sent and the server is started. Flask is used to manage this request of the administrator.

Flask is a BSD licensed microframework for Python. It is based on Werkzeug and Jinja2. It figures out what the requests are, what is being asked and what response is needed to send back to the administrator. Flask framework was selected because of its simple process of web application development. It allows focus on the administrator request and response in a seamless manner

5. SCORE GENERATION

The process of monitoring each screenshot manually is a bit tedious in a large LAN. A scoring system is used to score each client machine thus removing the need to check all the screenshots manually. These scores are calculated based on the work performed on the client machines in comparison with the work assigned to them by the administrator.

In order to calculate the relevance of the data, the administrator inputs keywords related to the assigned task. These keywords are then matched with the text data generated after applying OCR to the client machine’s screenshot. The score is generated by counting the number of keywords that occur in the text data generated from the screenshot. Higher the occurrence, the higher the score.The screenshots of the client machines when some software applications are running in the foreground do not produce meaningful text data which can be matched with the keywords. The list of running applications is checked for such applications and changes in the score are made accordingly.

In order to give the administrator complete information regarding the client’s work, the scores from multiple screenshots of the client machine over a period of time are calculated and stored. Weights are assigned to the scores based on the time it arrives at. The weight is higher for scores generated from the latest screenshots.

The formula of score calculation is given by:

$$\text{Net Score} = \frac{\sum_{i=0}^{n} Wi * Si}{n}$$

Where,

W = weight assigned to the score

S = Score generated from screenshots

n = Number of screenshots sent till calculating the net score

The clients are classified using a color-coded interface which exploits the ease of understanding of colors by a human brain thus making a clean and convenient UI.

Table 1 shows the scores generated for each keyword set against a particular screenshot. The screenshots used are of the following:

Idle Desktop, Google search for machine learning, YouTube search for machine learning, Wikipedia search for machine learning, Google search for Wikipedia

Thus from the following observations we can conclude that the relevancy score is most reliable for a rich set of keywords that are closely related to the task assigned. Moreover, it is observed that generic words like Google, YouTube, and Wikipedia might lead to unreliable higher scores for irrelevant images. Using larger sets of keywords won’t necessarily increase the score as all of the text might not be captured through the screenshots

6. USE CASES

6.1 Educational Institutions

This application is targeted to be used by college faculty while conducting laboratory sessions. This will provide an easy alternative to the manual assessment of students’ performance in the laboratory. This application can be used to send and install software in student’s computers through LAN, thus saving the time of individual manual installation. Online attendance systems can also be incorporated, thus automating the laboratory experience further.

6.2 Corporations

This application can be used to monitor the activities of multiple employees. It can be useful for ensuring that they are committed to their work. The software can also be tweaked to detect any unauthorized external devices connected to the employee machines.

Generally, in corporations, most of the suspicious services are denied access. In some exceptional cases, such services are needed. The proposed system provides an alternate approach to such situations. The corporations can block the surely harmful services and allow access to some suspicious services that are needed, as the administrator holds the power to terminate them for someone who tries to abuse the access.

6.3 Online Examination Portals

This application can be used as a plug-in for online examination portals to ensure that students attempting the exams do not cheat hence providing a fair environment for all the exam takers.

Table 1: Score generation based on different sets of Keywords for different screenshots

List of Keywords	Screenshots				
	Desktop	Google	YouTube	Wikipedia	Google homepage
Google, Wikipedia, YouTube, machine, learning	0	40	36	37	10
neural, network, regression, machine, learning	0	44	41	43	0
neural, network, regression, epoch, tensorflow	0	0	0	2	0
neural, network, regression, epoch, tensorflow, machine, learning, unsupervised, supervised	0	40	36	45	0
machine, learning	0	33	35	37	0

Table 2: Comparison of network monitoring and administration projects [4]

Features	Network monitoring and administration projects					
	Master BLACK (proposed s/w)	ActivTrak [5]	Amelia [6]	SolarWinds - Network Performance Monitor [7]	Wireshark [8]	PRTG [9]
Open source	-	-	✓	-	✓	-
Free	-	✓	✓	-	✓	✓
UI based on	Web	Cloud	Windows Application	Web	Window / Mac application	Windows/Mac/Android application
Screenshot sharing	✓	✓	✓	-	-	-
View Client IP, port	✓	-	✓	-	-	-
View running applications	✓	-	-	-	✓	✓
Video Playback, Alarms, Website blocking	-	✓	-	-	-	-
Network performance	-	-	✓	✓	-	-
Traffic, packets, bandwidth	-	-	-	-	✓	✓
Cloud services, Databases	-	-	-	-	-	✓
Client work-relevancy score	✓	-	-	-	-	-
Method	Socket Programming, OCR	-	java.net API via TCP	SNMP	Packet Sniffing	SNMP version 1, 2c 3
Use Case	Laboratory Session Monitoring	Insider Threat Detection, Organizational Efficiency, etc.	General Network Monitoring	Advanced Network Alerting	Network protocol analyzer	Infrastructure Management, Network Monitoring

7. COMPARISON WITH EXISTING PRODUCTS

Numerous software applications have been developed on the concept of LAN Monitoring. Each has its own features focusing on different aspects of monitoring. Table 2 shows a comparison of a few features of some of the existing LAN monitoring and administration products.

8. CONCLUSION

Software projects for monitoring devices over a network are available, but they tend to occupy human resources for constant monitoring. The score generation technique has proved to be quite accurate when given a smart choice of keywords. This allows the administrator to check the system at any point in time and obtain a concise observation of the relevance of every client's work throughout the time span. Thus, the application saves the time of monitoring each client's machine individually.

Furthermore, the current administration methodologies rely heavily on denial of access to suspicious services and generally provide information that is difficult to comprehend for a naive user. The features provided by the proposed system are intuitive and straightforward providing a raft of useful options for monitoring as well as administering devices over LAN. The system stands suitable for educational institutions and can be readily scaled for deployment in the corporate sphere.

9. FUTURE SCOPE OF APPLICATION

The software can be hosted on a mobile platform to introduce portability. Basically, an application can be developed to keep an eye on devices over a network from any remote GPRS enabled device [1][2]. Additional features that can be added are storing the live screenshots to examine them if required and installing software concurrently on all connected machines.

In order to further improve the efficiency of the scoring system, documents related to the work to be performed can be used to extract a rich list of keywords.

10. REFERENCES

[1] Karishma Gidge, Kalyani Patil, Priyanka Wadnere, "Android Based LAN Monitoring", Journal for Research in Applied Science & Engineering Technology (IJRASET) 3 (January 2015).

[2] Meghana Sapkal, Shekhar Patil, Leesensa Vispute, Santosh Jagtap, "GPRS Based LAN Monitoring and Controlling", IOSR Journal of Computer Engineering 16 (May-Jun 2014), pp. 09–15.

[3] Harsh Mittal, Manoj Jain, Latha Banda, "Monitoring Local Area Network Using Remote Method Invocation", International Journal of Computer Science and Mobile Computing (IJCSMC) 2 (May 2013), pp. 50–55.

[4] Paul Ferrill. The Best Network Monitoring Software of 2018. URL: https://in.pcmag.com/cloud-services/97759/the-best-network-monitoring-software

[5] ActivTrak, https://support.activtrak.com/hc/en-us

[6] Amelia, https://github.com/Obsidiam/amelia

[7] SolarWinds - Network Performance Monitor, https://support.solarwinds.com/@api/deki/files/10232/NPM_Administrator_Guide.pdf?revision=22

[8] Wireshark, https://www.wireshark.org/docs/wsdg_html/

[9] PRTG, https://www.paessler.com/manuals/prtg