MALWARE ANALYSIS USING DYNAMIC APPROACH

V. S. JEYALAKSHMI
Centre for Information Technology and Engineering,
Manonmaniam Sundaranar University,
Tirunelveli, India.
vsjeyalakshmiap@gmail.com

N. KRISHNAN
Centre for Information Technology and Engineering,
Manonmaniam Sundaranar University,
Tirunelveli, India.
krishnan@msuniv.ac.in

T. ARUMUGA MARIA DEVI
Centre for Information Technology and Engineering,
Manonmaniam Sundaranar University,
Tirunelveli, India.
arumugamariadevi@msuniv.ac.in

***Abstract*—Often, people call anything that corrupts their computer as a virus without knowing what it actually means or accomplishes. Sometimes malware steals all the business as well as individual's sensitive information and make severe loss. Ransomware, spyware, Trojan horse are some examples of malware. A lot of malware samples are obtained by anti- malware companies. Malware is run in a controlled environment using automated analysis tools to generate a report outlining the behavior of the program. The proposed study provides a systematic introduction to the different types of samples that fall under the broad blanket of malware, their unique characteristics, prerequisites for malware analysis, and the summary of the dynamic malware analysis tools and its process.**

Index Terms—*Dynamic malware analysis, virus, Anti-malware, tools, Reverse engineering.*

I. INTRODUCTION

In a real world, cyber criminals are inventing the viruses and launching daily for business purpose, so battle against viruses is a continuous processes which will never end. Cyber criminals are attackers and hackers in nation as well as states. Viruses make damages to the server, host system sometimes the whole network, etc. Hackers are always hiding threats in a sandbox that will not run until some conditions are met. Threats are very expensive in both quality and quantity. So researchers, academicians perform malware analysis and understand the working of malware (i.e) what are all the new characters exist in the malware. Sellers of software solutions give bulk amount as prizes for finding new threats, bugs which will help them to defend against malware attacks. The plan of this paper is to detect the malware working in a system using some tools [1], [2]. Three different kind of malware analysis exist: static, dynamic and hybrid. Static analysis extract the features without executing the emulator Identify applicable funding agency here. If none, delete this.or device. Signature based and permission based analysis are the two types come under static analysis. In signature based analysis, semantic patterns are filtered to identify the malware. The major drawback is if some current malware variants are not updated it can't be identified there. In permission based analysis, if the accessing rights are not fully given for any application then it will not be able to access all the resources. The drawback is only the manifest file is analyzed but not all the files. Analysis of static malware has some restrictions such as all the programming languages are not supported by automated tools, gives false positive and false negative, still now good trained personnel are not have to conduct the static code analysis, etc [1], [2]. Dynamic malware analysis discovers threat by scanning in runtime environment, there are many tools available. There is a chance to confirm static code analysis result and do not need to access the actual code, etc. So it is advisable to use dynamic malware analysis [1], [2]. Anomaly based detection and taint analysis are the two types of dynamic analysis. Anomaly based detection detects the abnormal traffic patterns like someone spying the system activity present in the network. Taint analysis is verifying the external data circulation as well as its working in the system. The combination of both the static and dynamic malware approach is called Hybrid type. It extracts the run time data from dynamic analysis and static algorithms are applied to detect the malicious functionality and its behavior of the threat [1], [2].

A. Motivation - Malware Analysis

Determining the mannerisms of a dangerous document or website is the procedure of malware analysis. As a result of the analysis, security problems are identified. Figure 1 shows the malware can be classified by a person with no superior knowledge of malware analysis are given according to the malware names. They are: I. Customary type classification; II. Classification related to code behavior; III. Classification related to Execution rights of the malware. Our entire system is made aware of attacks early on thanks to malware analysis. This study emphasizes the use of multiple tools for dynamic malware assessment.

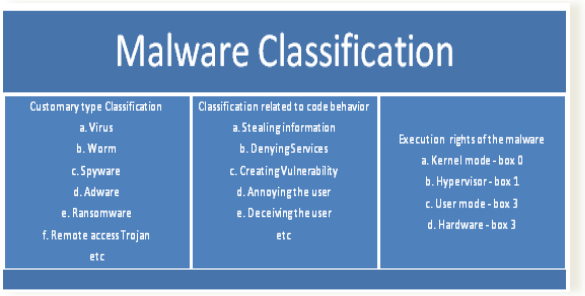

Fig. 1. Different Malware Taxonomies

The paper includes: 1. This study concentrates on using dynamic malware analysis to understand the features of malware. 2. To explain each dynamic malware analysis tool's methodology and application. 3. Reverse engineering tools and its characteristics.

This paper arrangement is as follows, Part I is dealing with malware behavior and analysis. The dynamic analysis characteristics and several dynamic malware analysis tool implementations are covered in Part II. The notion of reverse engineering and the tools used in it are introduced in Part III. In Part IV, the conclusion is presented.

II. BACKGROUND STUDY

A. DYNAMIC MALWARE ANALYSIS

In Dynamic malware analysis [1, 2, 3, 4, 5] to discover the malware behavior and its intentions by running the malware samples in Virtual machine or Sandbox. Sandbox is an isolated environment to run the malicious code from unknown attachments and observe its behavior without affecting the local applications before deployment. API calls, registry keys, network traffic, disk usage patterns, etc are collected by a virtual machine accompanied by monitoring software like cuckoo, PeStudio, Procmon, etc.

B. DYNAMIC MALWARE ANALYSIS PROS AND CONS

The suspected dangerous code is run throughout a dynamic malware analysis in a safe sandbox environment. Security experts can detect the malware in motion using this closed system without having it infect their computers or let it get into the local network. Threat hunters and incident responders can better understand the threat's true nature thanks to dynamic analysis, which gives them a deeper perspective. The time is needed to alter a file in order to find harmful code is therefore eliminated by automated sandboxing. Dynamic analysis presents a problem since opponents are knowledgeable and skilled at locating sandboxes because they are aware of their existence. Adversaries hide code that is dormant unless specific criteria are satisfied, at which point the code only runs [1, 2, 3, 4, 5] in order to spoof a sandbox.

C. DYNAMIC ANALYSIS TOOLS

(i) Procmon: Procmon is a Process Monitor (amazing tool) used to monitor all the registry entry, windows file system and real time process activity, etc. It records local system interactions. It is a free monitoring toolkit of malware hunting such as capturing the session IDs, usernames, logging information, consistent process information, etc [6,7]. It is like a setting of filter that captures all the needed details of each process activity. Fig 2 represents how the process monitor helps to identify the time when the activity of each process is handled, process name, process ID, the path where it locates, process result of each activity like success or deny each processactivity.

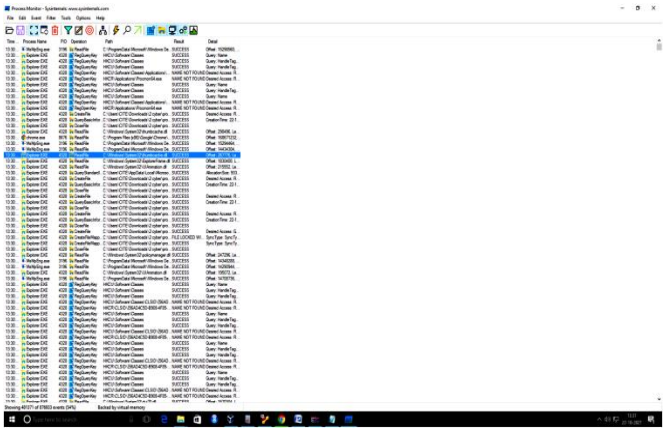

Fig. 2. Process Monitor

Event viewer shows all the log of each application, driver, reloaded events, installed program events and windows system messages including errors. Fig 3 represents to monitor the summary of administrative events like recently reloaded events and the log overview.

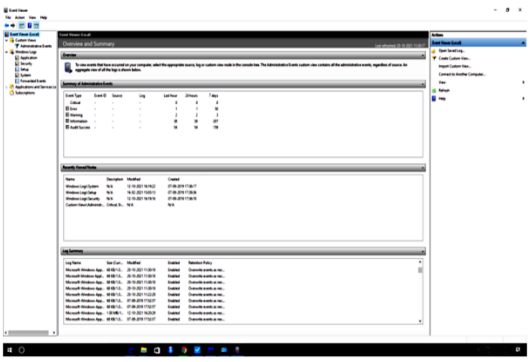

Fig. 3. Event Viewer

(ii) Process Explore: Process Explore, an open source tool manages and track all the information about what are all the process handled by the user or DLL have opened or loaded for each process in that system, which specific file or directory open in a user's system[8]. Figure 4 shows all the information about the process activity like PID, process size in bytes, description of each process, whether the CPU is working or suspended, the company name of each application.

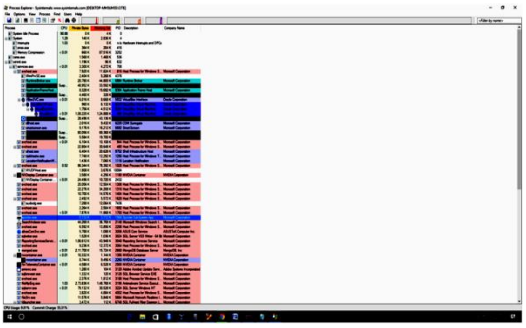

Fig. 4. Process Explore

(iii) Regshot: Regshot, a free tool used to take the snapshot of registry entry which is used for later comparison with another registry entry snapshot. It is used for identifying the system changes when a new software applications are installed [9]. Figure 5 represents the installing and scaning of Regshot comparison. Fig 6 shows the changes happened by any new product or application is installed.

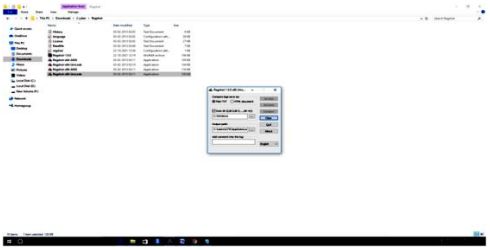

Fig. 5. Regshot

Fig. 6. Regshot View

(iv) ApateDNS: ApateDNS controls and gives the log information of all the DNS quries and DNS responses of all the host name, IP address,etc. Malware normally tries to connect the multiple hosts can be rectified by ApateDNS using NXDOMAIN[10]. Fig 7 reprsents the DNS responses of the user system.

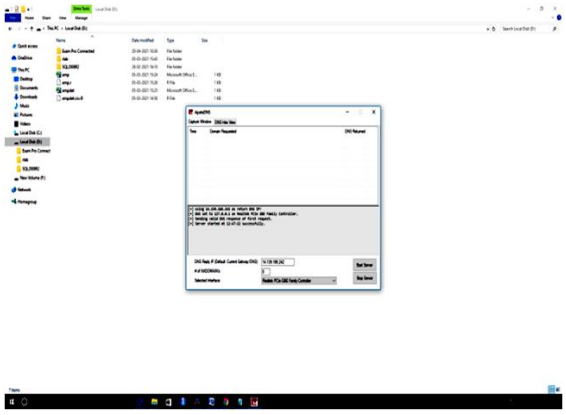

Fig. 7. Apate DNS

(v) Wireshark: Wireshark is a tool for monitoring the network traffic which is present/absent and all the network standards [11]. It is one of the best network protocol analysis tools which is used to analyze and capturing what is happening in the packet of file in the network. Clearly, Source and Destination IP address is used to identify that as whether suspicious or not. Figure 8 shows the wireshark capturing window with details about adapter, Ethernet, virtualbox host, etc. Fig 9 gives the information about Ethernet like source, destination, time, protocol, length information, etc.

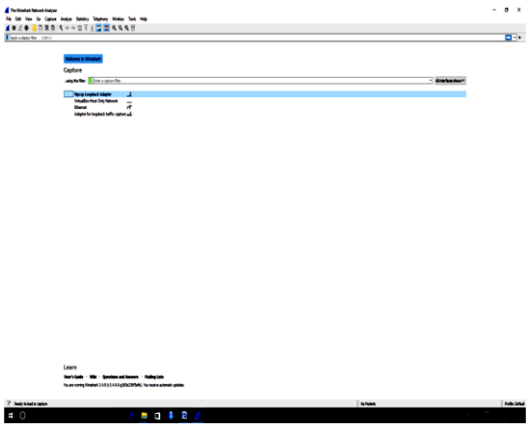

Fig. 8. Wireshark

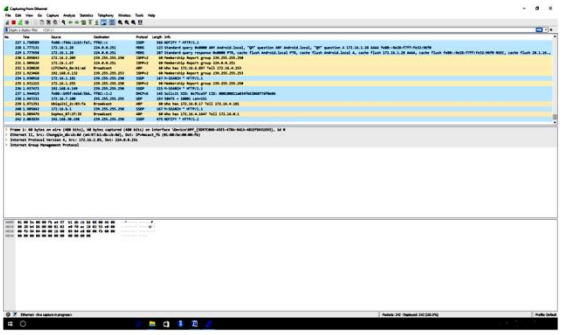

Fig. 9. Wireshark capture from Ethernet

(vi) PeStudio: PeStudio is a tool used to collect the windows executable binary for malware investigation whether the XML file is trustworthy or not [12]. It gives every single detail of an analysed executable file like hashes, packer identification, imports and exports, strings, etc. Figure 10 indicates the file hash values, libraries, import-export details, file type, file size, description, compiler time, debugger time, etc.

Once the hash value is identified from PeStudio, copy the sha256 - hash value and verify that as malware or not by paste the hash value in Virustotal.com and click search. If no matches are found then our file is good otherwise it is malware. Similarly, extract the strings from PeStudio, if strange is looking strings – random looking characters are found then it is more suspicious. In libraries, only one dll is found it is more suspicious. Imports, have to check the functions, there only one function is available means it is more suspicious. It detects the static property anomalies. IOC - Indicators of Compromise is a kind of signature type malware easy to identify by hash values. These are all initial behavior dynamic malware analysis.

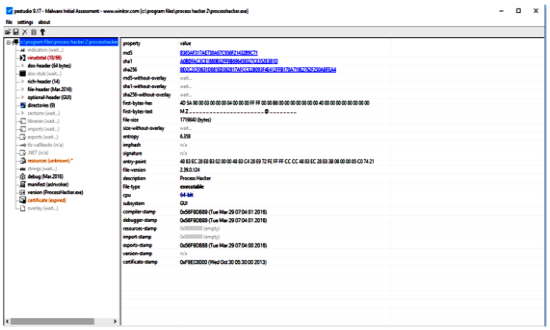

Fig. 11. Pe-Studio

(vii) Autoruns: It gives the information about startup monitor's auto-starting locations, like when the user switch on the system it boots up, when he/she logs in to a system in which the applications are set up to execute, autoruns provides information on toolbars, browsing assistance objects, auto-start services, and many other things. If a device is hacked, the malware that has been installed will also need to be able to reboot the system. The malware needs to be running on a machine even after a system is switched off in order to take use of genuine windows capabilities that enable the software to start up when the computer boots. The fact that virtually all malware is built to run automatically increases the likelihood that it may be found and eliminated with the support of auto-runs. All of the windows features are shown in Fig 11[17].

Fig. 10. Autoruns

Table. 1. Dynamic Analysis Tools and its Usage

S.No.	Tool Name	Period	Author	Purpose	Efficiency	Tool - Type	Public Usage
1.	PeStudio	2009 - 2022	Marc Ochsenmeier	Perform malware analysis on executable files	Fix crash, bugs when handling import address and exception table, better detection of file signature, etc	Open-source	Academic Research
2.	Procmon	2006	Mark Russinovich, Bryce cogswell	To monitor File, registry and system process activity	To capture file system, registry key activity, threat and network activity	Open – source	Academic Research, Real time
3.	Process Explorer	2001	Winternals Software	To collect information about all the system's process activity	To view all the CPU process activity, DLL, GPU, etc	Open – source	Academic Research, Real time
4.	Regshot	1999	TiANWEi	To compare the registry usage in windows	To check quickly the system changes between two different points in time	Open – source	Academic Research, Real time
5.	Comodo	2021	Comodo	To prevent breaches	To give endpoint protection from millions of threats like zero day exploits	Paid	Real time
6.	ApateDNS	2014	Ana Marculescu	To respond all DNS queries it processes	To capture the spoof DNS responses for a user specified IP address	Open – source	Academic Research, Real time
7.	Wireshark	1998	Gerald Combs	To analyze the Network troubleshooting, time consuming	To analyze the TCP and UDP traffic in the network	Open – source	Academic Research, Real time
8.	MCAfee	2021	John David	To protect from phishing, viruses, hackers	To alert from before connecting to risky websites	Paid	Real time
9.	Autoruns	2018	Mark Russinovich, Bryce Cogswell	To fix the minor bug when potentially unwanted software installed	To give the file and registry locations in a system when the system autostarts	Open – source	Academic Research, Real time
10.	Cuckoo Sandbox	2010	Claudio Guarnieri	To monitor the behavior of malicious process run in an isolated environment	It traces API calls, network traffic, malicious files, etc	Open – source	Academic Research, Real time
11.	Intezer Analyze	2020	Intezer	To classify the threats and give clear solutions	To analyze the files, URLs, system activity, etc	Paid	Real time
12.	ProcDot	2017	Christian Wojner	To monitor the system activity and fix the bug	To view the details due to an issue in a system, footprints of malicious software, etc	Open – source	Academic Research, Real time

(viii) Cuckoo Sandbox: Cuckoo is a free software program that runs automatically, examines files, and gathers exhaustive analysis data that describes what the virus does when it is functioning in isolation. The sandbox will keep track of the malware's actions and then generate a report on what the virus tried to accomplish while operating in this safe environment. It gathers IOCs fast that might be needed for a security event and provides immediate, in-depth details on how the virus is likely to act. Cuckoo has the ability to examine a wide variety of harmful files, including executables, word docs, Pdfs, emails, and hazardous scripts [18].

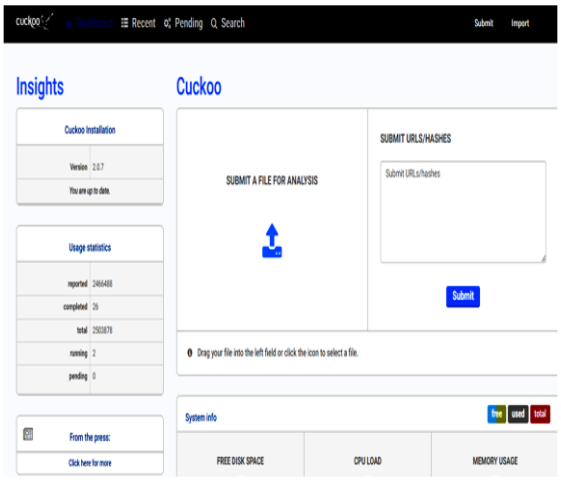

Fig. 12. Cuckoo Sandbox

Fig 12 shows online Cuckoo Sandbox tool. Use a tool to evaluate the supplied file and submit samples, hashes, and URLs. The dashboard contains some basic data about recently submitted files. Checking the APP calls and running the malware in a cuckoo sandbox can assist identify whether the malware is active or not.

Fig 13 contains information such as file sizes and hashes. A score range from zero indicates the file is benign and growing to ten indicates the file is more malicious. Signatures are color-coded. Blue signature denotes benign, yellow signature denotes medium risk and the red signatures are more malicious. Cuckoo gives a screenshots contains the information about the malware infected Guest machine. Cuckoo records any domains IPs that have been recognized which is used to distinguish other compromised hosts within an association. Proactive blocks can also be set up to protect any other hosts from dealing with these malignant IP addresses.

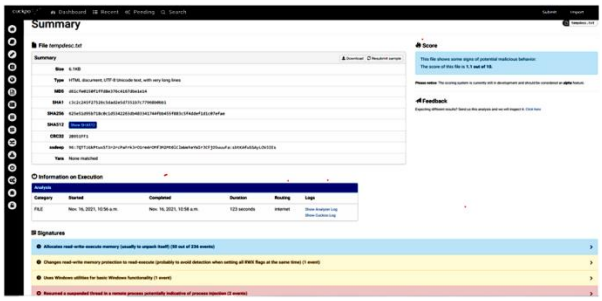

Fig. 13. Cuckoo Sandbox Summary

III. Reverse Engineering

Reverse Engineering is the system of generating the code from the structure of the product which is indicated in the Figure 14. It involves collecting and examining the information, functionalities, the control flow, the data flow and many others.

Fig. 14. Reverse Engineering

A. Reverse Engineering Tools

- IDA Pro: It is a disassembler to make an interpretation of machine code into an intelligible configuration (low level computing – assembly language). It is utilized to gather the guidance follow logs of executables. Assembly language is exceptionally difficult to read. Further examination should be possible in the guide of the program code to distinguish the malware [13].
- Apktool: It is a reverse engineering tool for Android apk files.
- OllyDbg: It is a 32 bit disassemble, debugger emphasis on binary code analysis even where source is unavailable. It is used to analyse the code in trace registers, API calls, DLL, recognize procedures and loops[14].
- WinAPIOverride: Software is used to monitor and extract API calls during its execution.
- Buster Sandbox Analyzer is utilized to investigate the processes behavior and the progressions made to the system like file system, registry and port to identify the malware [15].

IV. CONCLUSION

From the above study, it is said that for detecting threats in a good way is by dynamic malware analysis. Security problems are created everyday and it was solved to save the society by making alerts. Malware investigation is to identify the threat behavior in the operating system of the computer and give security to the whole environment [15, 16]. New analysis tools were detected and updated by Cyber Securitist. Different dynamic malware analysis tools and its outputs are discussed well in this paper. Latest different malicious actions are practically detected by the dynamic malware analysis tools in future work.

References

[1] [Available:[https://www.ijedr.org/papers/IJEDR170223.pdf
[2] Available:[https://gcn.com/cybersecurity/2009/02/static-vs-dynamic-code-analysis- advantages-and-disadvantages/287891/]
[3] Available:[www.crowdstrike.com/cybersecurity-101/malware/malware-analysis/]
[4] Introduction to malware analysis - Available: [https://quickheal.com]
[5] Dynamic Malware Analysis in the Modern Era – A State of the Art Survey, Orior-Meir, Nir Nissim, Yuval elovici and Lior rokach. https://doi.org/10.1145/3329786
[6] Procmon - Available :[https://docs.microsoft.com /en-us/sysinternals/downloads/procmon]
[7] Event viewer - Desktop Window
[8] Process explorer–Available:[https://docs. microsoft.com/enus/sysinternals/downloads/ process-explorer]
[9] Regshot – Available:[https://sourceforge.net /projects /regshot/]
[10] ApateDNS – Available:[sdl-apatedns (WinRAR file)]
[11] Wireshark - Available:[https://www.wireshark. org/]
[12] Pestudio – Available:[https://www.winitor.com
[13] download/]Available: [https://www.educba.com/reverse-engineering-tools/]
[14] Malware Detection with Sequence-Based Machine Learning and Deep Learning, William B. Andreopoulos
[15] A Close Look at a Daily Dataset of Malware Samples, Xabier Ugarte-Pedrero, Mariano Graziano, Davide Balzarotti.
[16] Available:[www.crowdstrike.com/cybersecurity - 101/malware /malware - analysis/]
[17] Available:[https://docs.microsoft.com/en-us /sysinternals /downloads/autoruns]
[18] Available: [https://docs.cuckoosandbox.org /en/latest /introduction/what/]
[19] Available: [https://www.comodo.com]
[20] Available: [https://www.mcafee.com]
[21] Available: [https://www.intezer.com]

Biography

V.S. Jeyalakshmi received her M.Tech degree in Computer & Information Technology from Manonmaniam Sundaranar University, Tirunelveli, India in 2010 Presently she is pursuing her Ph.D degree at Centre for InformationTechnology and Engineering of Manonmaniam Sundaranar University, Tirunelveli, India. Her research interests include Pattern recognition, Cyber Security, Big Data Science & Analytics, Machine Learning and Deep Learning.

Nallaperumal Krishnan received M.Sc. degree in Mathematics from Madurai Kamaraj University, Madurai, India in 1985, M.Tech degree in Computer and Information Sciences from Cochin University of Science and Technology, Kochi, India in 1988 and Ph.D. degree in Computer Science & Engineering from Manonmaniam Sundaranar University, Tirunelveli. Currently, he is the Professor and Head of Center for Information Technology and Engineering of Manonmaniam Sundaranar University. His research interests include Signal and Image Processing, Remote Sensing, Visual Perception, and mathematical morphology fuzzy logic and pattern recognition. He has authored three books, edited 18 volumes and published 150 scientific papers in Journals/proceedings/books and has produced 35 Ph.D. Scholars. He is a Senior Member of the IEEE and chair of IEEE Madras Section Signal Processing/Computational Intelligence/Computer Joint Societies Chapter.

Dr. T. Arumuga Maria Devi Received B.E. Degree in Electronics & Communication Engineering from Manonmaniam Sundaranar University, Tirunelveli, India in 2003, M.Tech degree in Computer & Information Technology from MS University in 2005 and Worked as Lecturer in department of Electronics & Communication Engineering in Sardar Raja College of Engineering and also received Ph.D Degree in Information Technology – Computer Science and Engineering from MS University in 2012 and also the Assistant Professor of CITE of MS University since November 2005 onwards. Her research interests include Signal and Image Processing, Multimedia and Remote Communication.