International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

Forensic Acquisition of IOS Devices

Hyndavi Koganti, G Siva Nageswara Rao

Abstract*: Apple devices are well known for their high-security features in terms of data storage. IOS devices have some restrictions for their usability. The device contains only internal memory and the users can back up their data into iCloud or iTunes. IOS devices are more secure when compared to other mobile devices. The IOS devices can also be jailbroken for the purpose of removing software restrictions and allows the installation of application from the unknown sources i.e., the app that are not unavailable in Apple App store. After jailbreaking, the device becomes vulnerable and lets the attacker to access the device. Apple provides both hardware and software patches to the vulnerabilities, which means many versions can't be jailbreak so easily. To perform Forensic investigation on the IOS devices, even the forensic investigators need privilege escalation to access the data of the device. The tools which are used to investigate IOS devices are avail as commercial. This project proposes an opensource method to access the IOS device using SSH shell. After the successful mount of device, the data can be acquired for further forensic analysis. Based on the artifacts analysed, the investigators can be able to find the root cause of the crime.

Keywords: iOS, plist, db files.

I. INTRODUCTION

IOS Introduction:

The first iPhone was formally launched and created accessible to the overall public in 2007, and this iPhone and was heralded noticeably at the Macworld of that exact same year. Throughout this first unhitch, the iPhone was accessible among the U. S. of America, UK, Canada, Germany, Norway, Sweden, Finland, France, Spain, European nation and African country. Jobs expressed his belief that pill PCs and ancient PDAs weren't wise choices as high-demand markets for Apple to enter, despite receiving many requests for Apple to create another personal digital assistant.

On January 9, 2007, Steve Jobs declared the first iPhone at the Macworld convention, receiving substantial media attention. Jobs declared that the first iPhone would be discharged later that year. On June 11, 2007, Apple declared at the Apple's Worldwide Developers Conference that the iPhone would support third-party applications pattern the campaign engine. Third parties would be able to turn out web 2.0 applications that users could access via the net. Such applications appeared even before the discharge of the iPhone; the first of these, mentioned as One Trip, was a program meant to remain track of users' trying lists. On June 29, 2007, Apple discharged version seven.3 of iTunes to coincide with the discharge of iPhone. This unhitch contains support for iPhone service activation and syncing.

IOS Architecture:

iOS is an iPhone operating system and it was created by Apple for the devices which are manufactured in Apple like iPhone, iPad, iPod, etc. Just like other operating system iOS has an architecture and it is a layered architecture.

In iOS architecture there are four different layers and these layers works as intermediate layer between the application and the hardware because the application and the hardware do not communicate each other directly. In this architecture a lower layer provides a normal service and the higher layer provides GUI services.

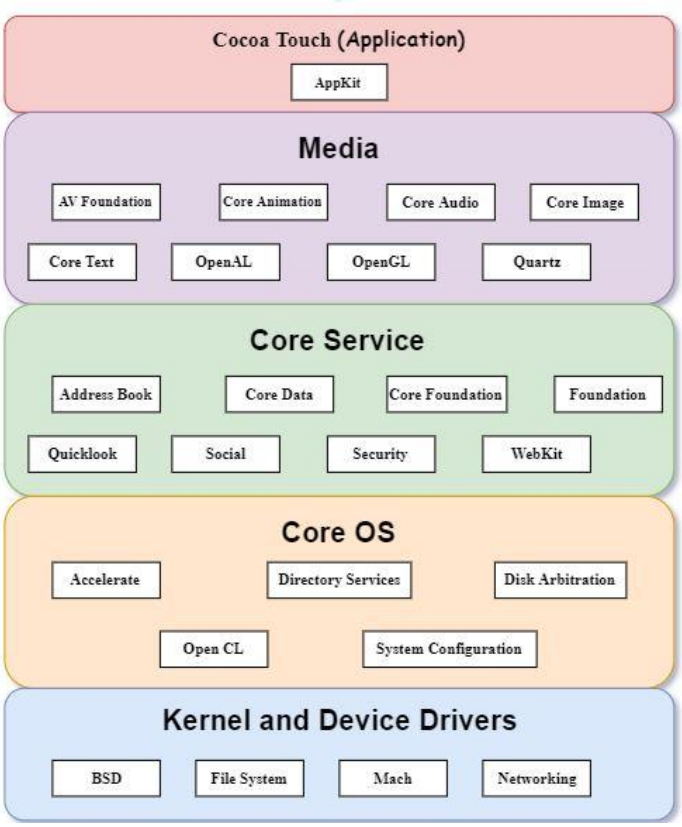


Figure 1: Architecture of iOS

1. Cocoa Touch Layer

The Cocoa bit layer is primarily answerable for the looks of apps. It provides access to main system functions like Contacts, Camera, bit input, shares with different apps, push notifications etc.

Manuscript published on November 30, 2019.
* Correspondence Author
Hyndavi Koganti*, M.Tech - Cyber Security and Digital Forensics, Department of CSE, KLEF, Vaddeswaram, A.P, India. Email: hyndavikoganti@gmail.com
Dr Siva Nageswara Rao, Associate Professor, Department of CSE, KLEF, Vaddeswaram, A.P, India.. Email: sivanags@kluniversity.in

Event Kit framework: This framework is an interface which helps to see and perform the operations related to the calendar.

Game Kit Framework: This framework helps to implement the support to the game centre to share the game related information in online.

iAd Framework: This framework helps and allows to deliver advertisements from application which based on a banner.

Map Kit Framework: This framework gives a scrollable map and that map can also be included into user interface of an application.

Push Kit Framework: This framework will provide a registration support for the application which support VoIP.

Twitter Framework: This framework is an UI which supports to generate tweets and also for creating URLs to access the services provided by Twitter.

UI Kit Framework: This framework is a vital infrastructure to applying any graphical and event-driven applications in iOS. And it has some important functions like multitasking, application management and infrastructure, cut, copy and paste operations and also supports touch and motion events.

2. Media Layer

The Media layer assist you to include second and 3D graphics, animations, image effects, and professional-grade audio and video functionalities into your mobile app.

Graphics Framework:

UI Kit Graphics: This helps to describe the higher-level support for the images which are designed and also for animated images to the content.

Core Graphics framework: This framework is a drawing engine for applications in iOS and it supports 2D vectors and images which are based on rendering.

Core Animation: This framework is an initial technology which optimizes the animation of application.

Core Images: This framework provides an advanced support to control video and motionless images and makes sure that the controlling the images in a non-destructive way.

Open Gl ES and GL Kit: This framework is an interface which was accelerated by hardware to manage 2D and 3D rendering.

Metal: This offers a very low overhead access to A7. And also permits to perform sophisticated graphics rendering and computation works in a high level.

Audio Framework:

Media Player Framework: This framework is one of the high-level frameworks and it gives and supports to easy use and access to iTunes library and playing playlist.

AV Foundation: This is an objective interface which handles the recordings and playbacks of audios and videos in the device.

Open AL: This framework is used to provide audio and it is a standard technology from the industry.

Video Framework:

AV Kit: This framework provides collection of user interfaces which are easy to present video.

AV Foundation: This framework provides an advanced capability for recording and video playback.

Core Media: This framework is used to describe the low-level interfaces and to describe data types for operating media.

3. Core Services

The Core Services layer comprises core services like address book, Security, Social and foundation which offer essential options to apps. It offers access to basic resources required for app.

Address book framework: This framework give access to a database which consists the contacts of the user in a programmatically way.

Cloud Kit framework: This framework provides a medium which helps to move data between the user app and iCloud.

Core data Framework: This framework is a technology which manages the Model View Control application's data model.

Core Foundation framework: This framework is an interface which gives fundamental data and service features to the application in iOS.

Core Location framework: This framework gives the information about geographical location and heading information of the applications.

Core Motion Framework: This framework can able to access the data which work based on motion of the device. Using this framework can also be able to access the information which works based on accelerometer.

Foundation Framework: This framework consists the features which were programmed through objective C.

Health kit framework: This framework consists and handles the health-related information of user.

Home kit framework: This framework can able to access the devices which are connected in the user's house.

Social framework: It is an interface for accessing the user social media accounts.

Store Kit framework: This framework supports for buying content and services from inside of iOS apps. Simply it is known as In-App Purchase.

4. Core OS

The Core OS layer comprises technologies and frameworks which offer low-level services associated with low-level hardware and networks. Core OS is the bottom layer and it work just like a foundation to iOS and maintains the other layers on the top of this layer properly. This layer is responsible for memory operations after completion of application like managing, allocating, releasing memory and takes care of the tasks of file systems, and other tasks in OS, and this layer can directly interact with hardware easily.

5. Kernel and Device Drivers

This is very cheap layer of iOS that in the main includes the kernel and device drivers. The kernel atmosphere is constructed on prime of Ernst Mach three.0 (a microkernel that replaces the kernel within the BSD version of Unix) and provides superior networking facilities and support for multiple, integrated file systems.

II. LITERATURE REVIEW

Mobile Forensics is the process of extracting Electronic or Digital evidence from mobile devices under Forensic conditions. These include iPad, iPhone, Smartphones, etc. While performing a forensic investigation, the forensic experts examine the evidence systematically with utmost careful so that evidence will not be modified or damaged.

The investigators will analyse only the particular data related to the case using forensic tools. The extracted data could be application files, SQLite database files, or keychains.

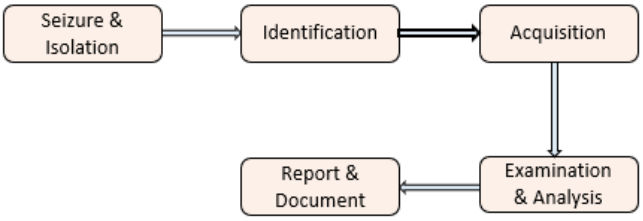


Figure 2: Data Extraction Phases

Seizure & Isolation:

The device should be disconnected from all the network and flight mode must be activated to avoid further communication to the device.

The device must be placed in the Faraday bag to prevent data tampering.

The simcard and memory card should be ejected.

Identification:

In what place does the mobile was identified?

What is the state of the mobile when available?

What data need to be extracted from the device?

Acquisition:

SSH connection should be established between the iPhone and computer to the same local network to maintain data integrity.

The backup of iPhone should be collected.

The passcode needs to be bypassed if passcode is present.

The Application files, SQLite database files, ktx files,.etc will be collected and stored.

Examination & Analysis:

The plist files, db files, ktx files will be analysed by using forensic tools and the data is extracted.

The deleted data will be recovered for evidence examination.

Report & Document:

The detailed document of the entire procedure, methods, tools used, the data retrieved will be prepared by the examiners which is a report as well as evidence.

The way to extract the complete data from the mobile device is by using Commercial Forensic tools. The other way of extracting data is possible with the help of jail breaking the device.

Jail breaking is the process of removing software limitations which are obtruded by Apple and performed on iOS devices to gain and attain the privilege escalation of the device. After jailbroken, the device will be ready to access all the data and collect the evidence which can be used in forensic examination.

III. APPLICATION STRUCTURE

IOS Application Structure:

Each and every iOS application operates in its own sandbox.

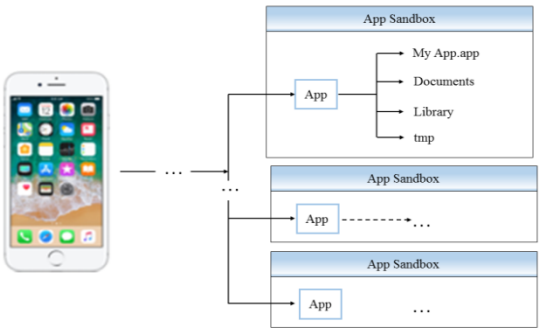


Figure 3: iOS Application Structure

Exception cases:

1. The public system interfaces like contacts, calendars, photo library.
2. Any application can use these interfaces provided when the user approves its access.

iOS Standard Directories:

- /AppName.app: This directory contains the app itself.
- /Documents: This directory stores user documents and data files of applications.
- /Documents/Inbox: This directory is used to read and delete files but it cannot create or write to the files which already exist.
- /Library: This is the top-level directory which contains files but not data files and it is used to create sub directories.
- /tmp: This directory is used to write temporary files that do not persist between the app launched. The files should be removed when they are no longer needed.

IV. DATA ACQUISITION PROCESS

Data Acquisition is the process of imaging the electronic evidence for examining the data present in the device. Extracting information from mobile device is not so easy.

Types:

1. Physical Acquisition:

It is the method which makes a bit-by-bit duplicate of a whole file system, comparable to approach taken in computer forensic examinations.

2. Logical Acquisition:

It is the method of extracting the logical storage objects which resides on the filesystem.

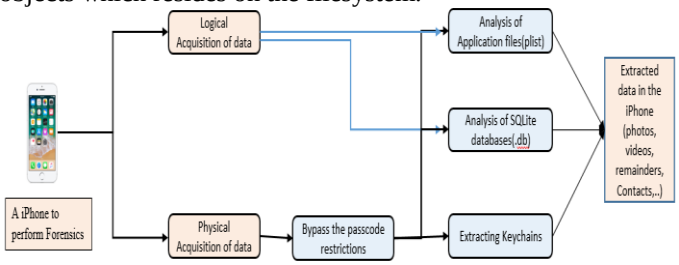


Figure 4: A visual architecture of my proposed mode

After the device is brought to perform forensic investigation, the data can be extracted either in the form of Physical or Logical acquisition methods. If there is passcode restriction, then bypass should be done to read the encrypted file system. Different types of file systems like plist(property list) files, SQLite (.db) Database files, Keychains(a database which stores sensitive information and encrypted with hardware key) will be present. Using different forensic tools, data in iPhone will be extracted.

V. PROPOSED METHODOLOGY

The IOS devices possess the potential shreds of evidences, which are acquired and analyzed for the forensic investigation.

This project proposes an open source approach to perform the forensic analysis of the IOS device. The device can be accesses through the Shell as it has linux kernel. The root of the device is accessed by using SSH shell and the data available in the storage is acquired for further forensic analysis.

The device can be accessed using the IP address of the mobile device through SSH port. Mobile devices store the system data, application data in the database file format. So to analyze the artifacts, those .db files or sqlite files and plist files are acquired. After gaining the shell access to the device, important artifacts like contacts, SMS, media etc., can be extracted.

In order to mitigate the cost effectiveness of existing system and to find artifacts in iPhone for forensics purpose, an open-source approach is designed.

VI. DATA ANALYSIS

Artifacts Extraction & Analysis

A piece of data that relevant to digital forensic investigation.

SMS: It is used for offline communication to send and receive message.

Sms.db: This database file which contains information about sent and received messages. The query used to extract the data is "**select * from message**."

Path: /private/var/mobile/Library/SMS/sms.db

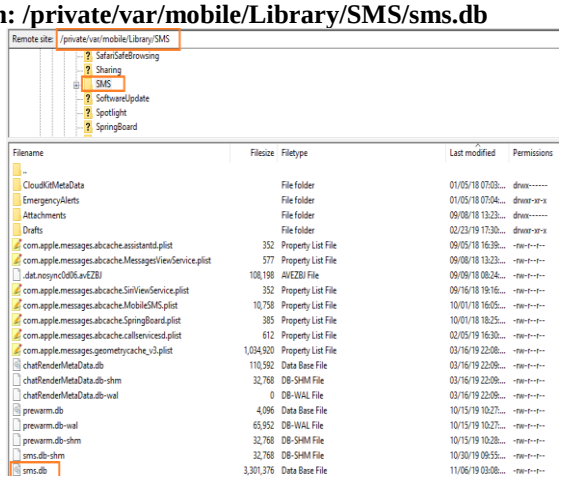


Figure 5: SMS Path

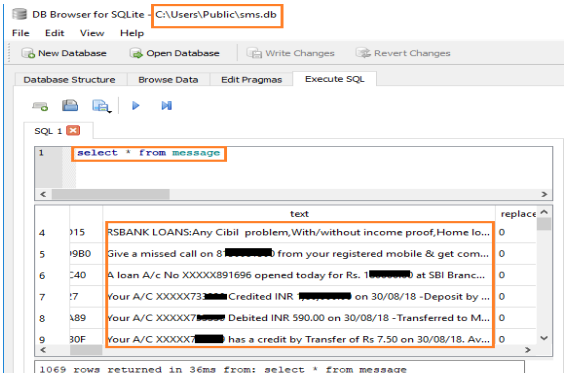


Figure 6: Data extracted from sms.db

Contacts: The entity which is created for identifying other person.

AddressBook.sqlitedb: This database file contains identify of persons which includes mail, Home, multiple storage of numbers. The query used to extract the data is "select * from ABPersonFullTextSearch_content"

Path: /private/var/mobile/Library/AddressBook

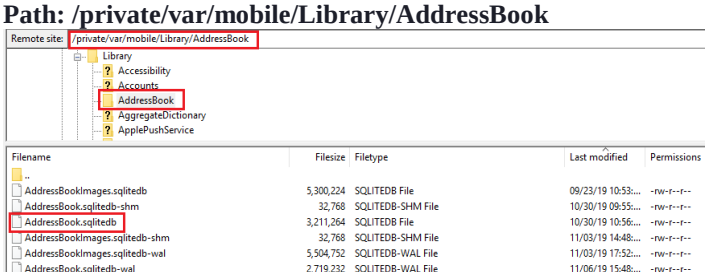


Figure 7: Contacts Path

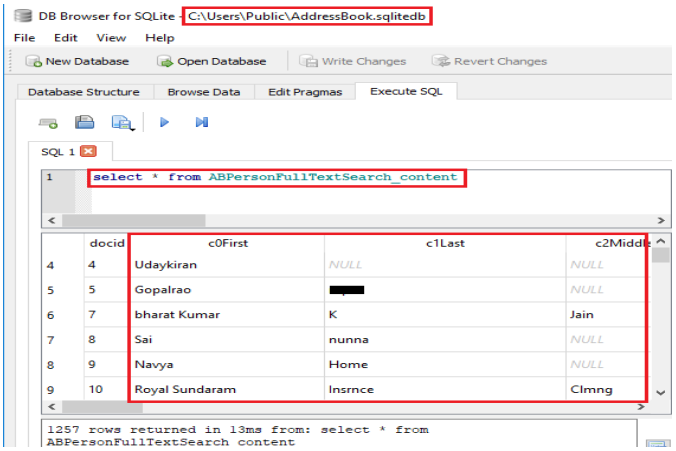


Figure 8: Contact names from AddressBook.sqlitedb

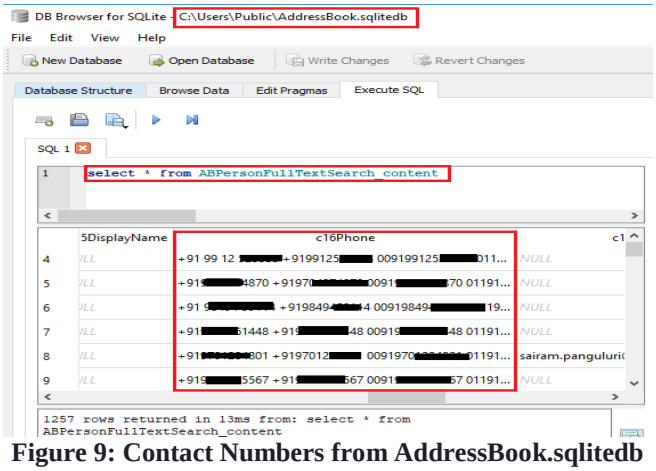


Figure 9: Contact Numbers from AddressBook.sqlitedb

Call Logs: It contains the statistical record information of owner calls in the form of logs.

CallHistory.storedata: This database file contains information about the call statistical record and Timings, Location, Duration. The query used to extract the data is "**select * from ZCALLRECORD**"

Path: /private/var/mobile/Library/CallHistoryDB

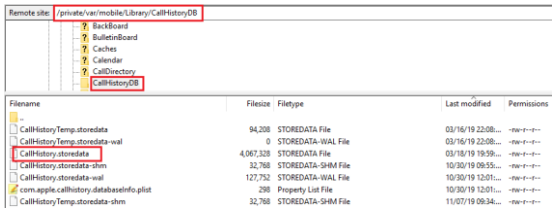


Figure 10: Call Logs path

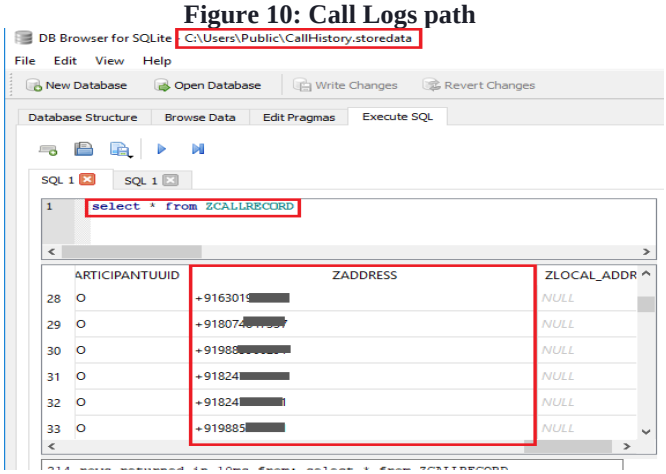


Figure 11: Phone Logs retrieved successfully

Accounts: It contains information about the device and its account holder.

data_ark.plist: This property list file contains information about the name of the system the last iTunes backup was taken and also the details of the iPhone.

Path:
/private/var/root/Library/Lockdown/data_ark.plist

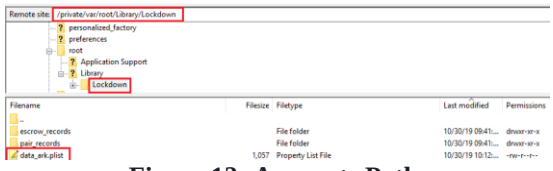


Figure 12: Accounts Path

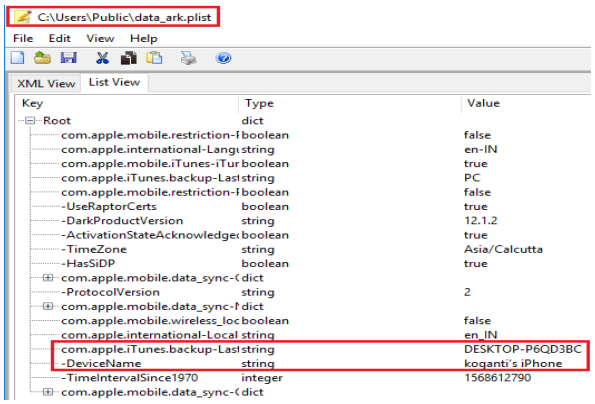
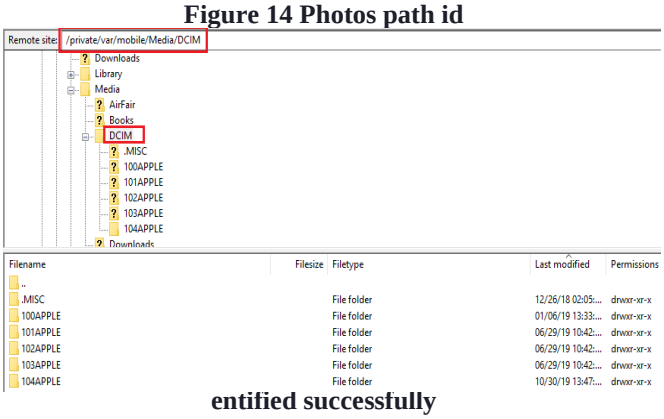


Figure 13: Device information from plist file

Photos: It contains user photos which are built via camera, screenshots and WhatsApp.

Path: /private/var/mobile/DCIM/

Figure 14 Photos path id



entified successfully

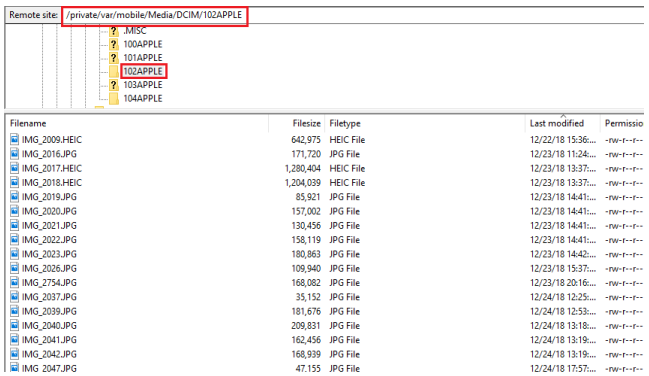


Figure 15 Images are successfully extracted

Passwords: These are used to secure the device or applications and sensitive data. Keychain-2.db file contains data which will be in encrypted form. The query used to extract the data is "**select * from synckeys**"

Path : /private/var/Keychains

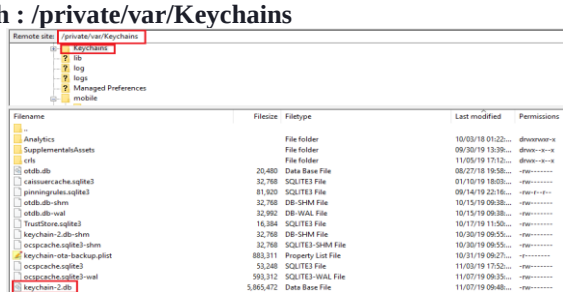


Figure 16: Passwords stored path is identified successfully

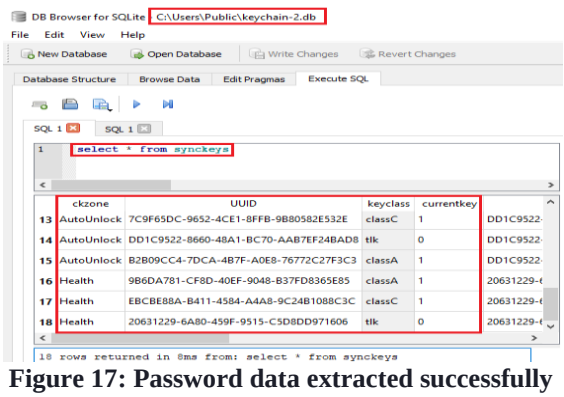


Figure 17: Password data extracted successfully

Third party applications:

These are the applications which are provided by vendors other than Manufacturers. iOS contains AppStore to install Applications onto the devices by secure authentication either to provide Apple password or finger access. The Names of the applications in the path are dynamic which means whenever it is newly connected for forensic purpose the names will be modified but not the data.

Path: /private/var/mobile/Containers/Data/Application

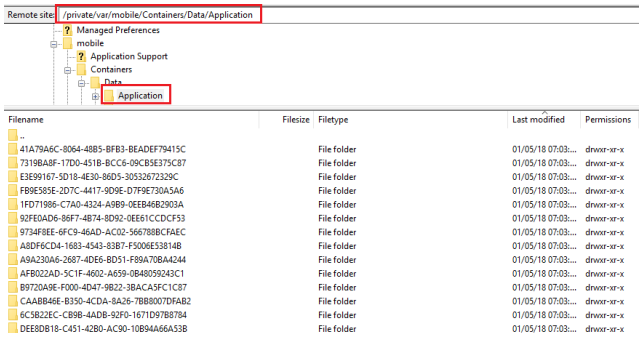


Figure 18: Vendor Applications Path

WhatsApp: It is a free messaging App which allows users to communicate and share data (photos, videos, GIF, Document, Location, Contacts, Voice Messages) with the help of Internet.

ChatMedia Folder contains media and status of different users.

Path:
/private/var/mobile/Containers/Data/Application/5E66E731-D1DA-4DE5-BC1C-127B44C00332/Library/Caches/ChatMedia

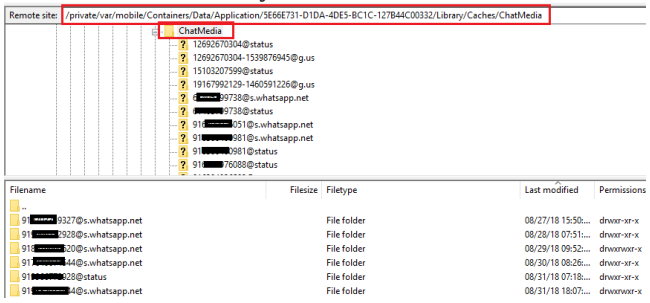


Figure 19: WhatsApp path

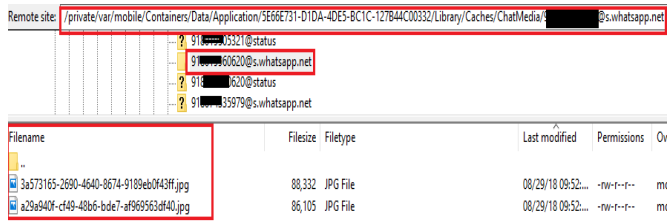


Figure 20: Data of One Person

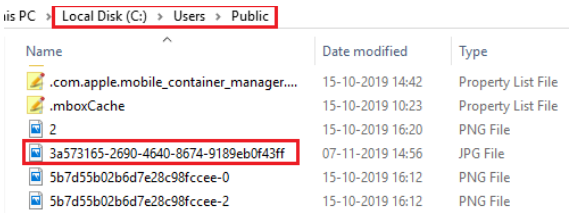


Figure 21: Extracted data of WhatsApp

Facebook: This is a free, cross-platform application which allows different people to make friends, to follow people, messaging, sharing images, and comment on photos or posts.

Recent_activites_notifications.txt file contains list of notifications received or popped-up.

Path:
/private/var/mobile/Containers/Data/Application/E0DB4DD7-7175-4153-AFBD-218A1C2CB5CA/Library/Caches

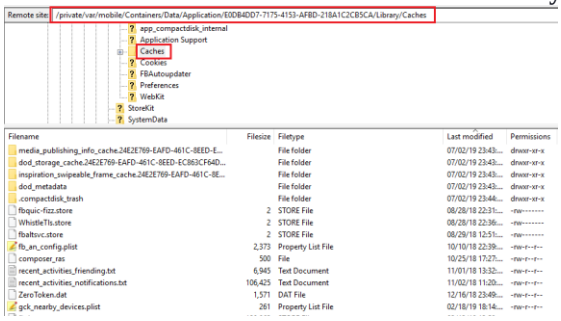


Figure 22: Facebook Path and data

Instagram: This Application is used to follow the activities or posts and send messages to different people.

Path:
/private/var/mobile/Containers/Data/Application/EA14ABB2-EFDB-4F9F-8E0B-A1D550590E22/Library/Caches

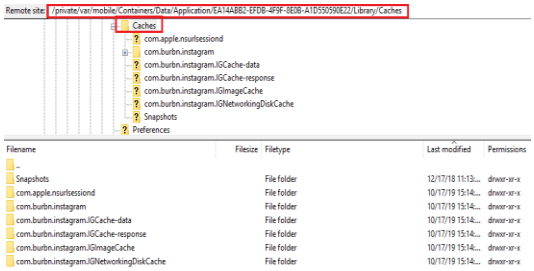


Figure 23: Cache Information

Path:
/private/var/mobile/Containers/Data/Application/EA14ABB2-EFDB-4F9F-8E0B-A1D550590E22/Library/Caches/Snapshots/com.burbn.instagram

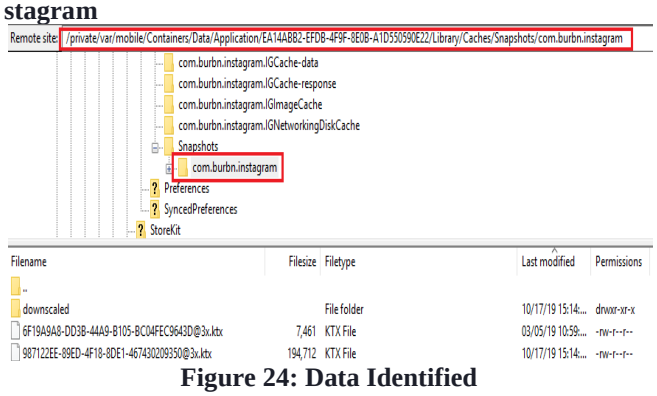


Figure 24: Data Identified

SBI YONO: This Mobile Application of SBI is used to perform monetary transactions, view account balance and place check book requests for bank.

Path:
/private/var/mobile/Containers/Data/Application/0CB96E66-E635-4654-99C8-051168ADD772/Documents/Inbox

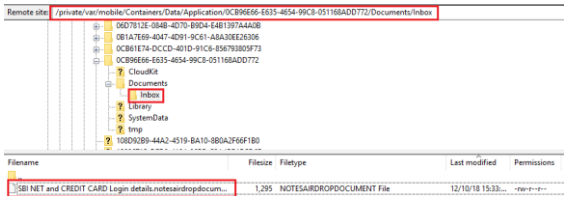


Figure 25: SBI YONO Path

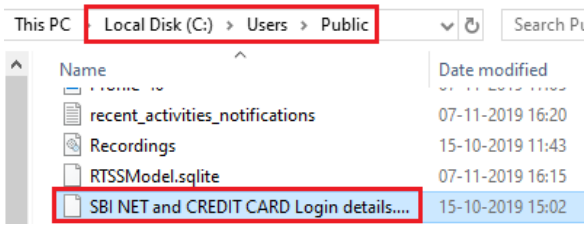


Figure 26: Information extracted

Axisbank: This Mobile Application of AXIS bank is used to transfer money from one account to another account with the help of account numbers, IFSC Code.

Path:
/private/var/mobile/Containers/Data/Application/6C7CCC94-0854-476B-AC2E-474024841254/Documents

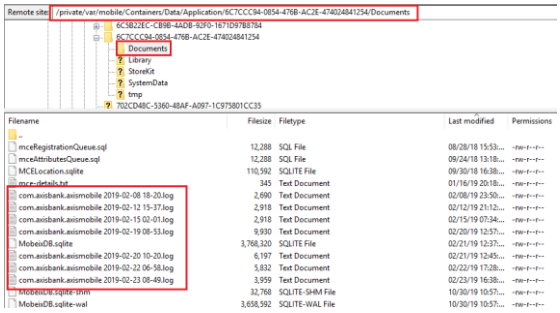


Figure 27: AXIS Bank path and log files

Flipkart: This Shopping Application is used to know the list of items which are searched by user and purchased. The information is stored in flipkart.sqlite.

The query used to extract the data is "**select * from ZCROSSPLATFORMCACHE**"

Path:
/private/var/mobile/Containers/Data/Application/C0E87A33-ABB6-480C-8BFE-DA2F78443F8E/Library

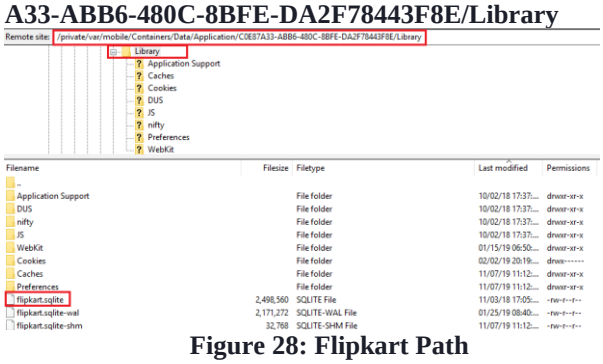


Figure 28: Flipkart Path

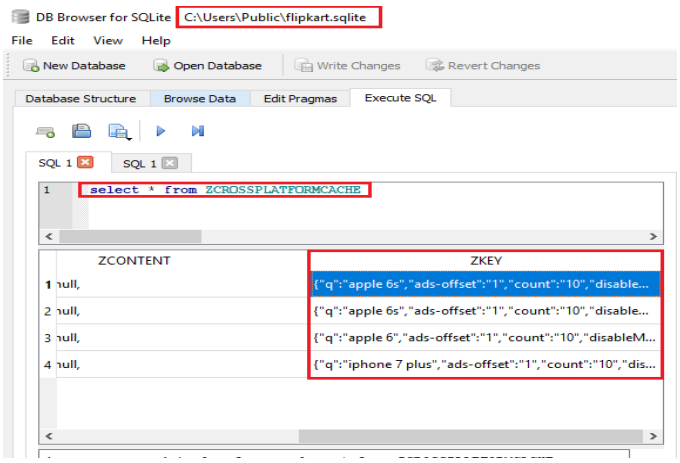


Figure 29: Searched Data of Flipkart

BookMyShow: This Entertainment App is used for checking Movies List, Date and Location for which the user has browsed.

Path:
/private/var/mobile/Containers/Data/Application/91AFD7FD-D91A-4363-BFF7-09B12F36D9FB/Library/Caches

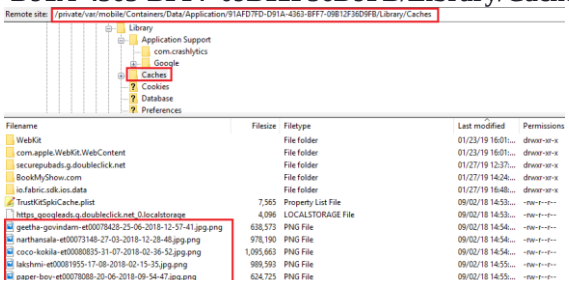


Figure 30: BookMyShow Path and Browsed Information

HPPrinter: This Application is used for printing documents or scanning Applications. Inbox folder contains the list of documents user has taken printout.

Path:
/private/var/mobile/Containers/Data/Application/06D7812E-084B-4D70-B9D4-E4B1397A4A0B/Documents/Inbox

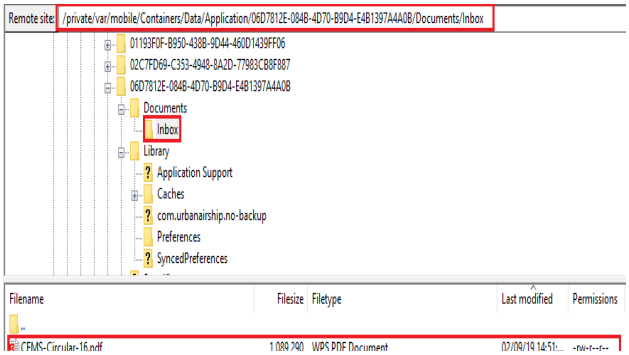


Figure 31: HP Printer Data and Path

FindMyiPhone: This Application is used in iPhones to find the location of the iPhone if the mobile is lost or missing. It requires internet connection and Location needs to be turned on.

FMIP.sqlite folder contains the locations of iPhone.

Path:
/private/var/mobile/Containers/Data/Application/6C5B22EC-CB9B-4ADB-92F0-1671D97B8784/Documents

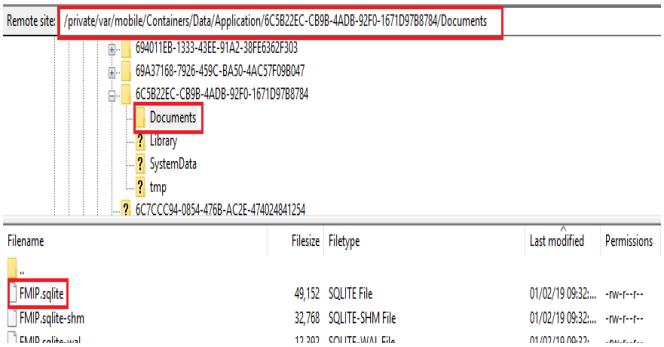


Figure 32: FindMyiPhone Path and database file

Google Maps: This route-map Application is used to find the Locations which are used by the user for travelling or searched.

Com.google.Maps.plist file contains the data when the user has used GoogleMaps.

Path:
/private/var/mobile/Containers/Data/Application/83B0F882-A1B8-4462-9937-B07B9D98ACB9/Library/Preferences

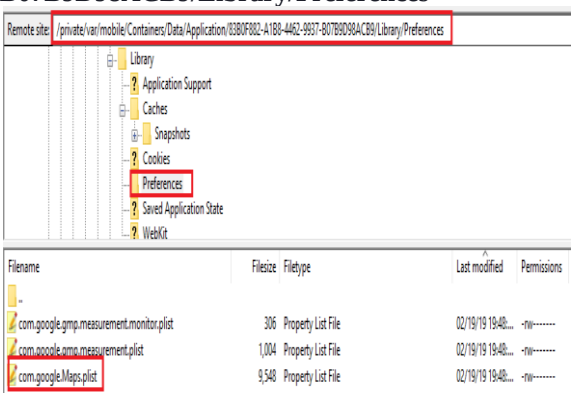


Figure 33: GoogleMaps path

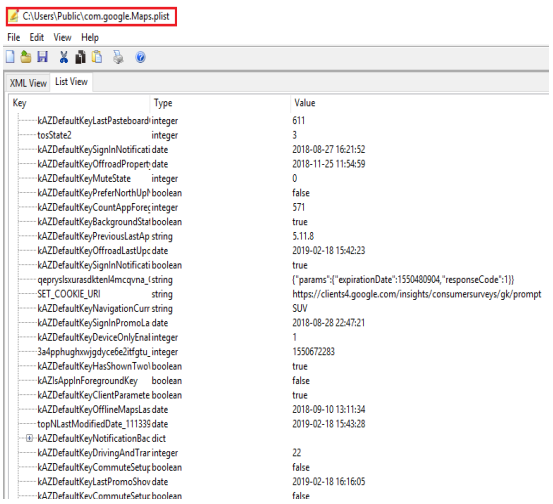


Figure 34: GoogleMaps data from plist file

BSNL: This Application is used to know the connection and registered user details.

BSNL.sqlite database: This file contains information of Service number, Account number and name of the user. The query used to extract the data is "select * from Items"

Path:
/private/var/mobile/Containers/Data/Application/93DD2AFF-86AE-41AD-85F3-43647ABB3AF6/Documents

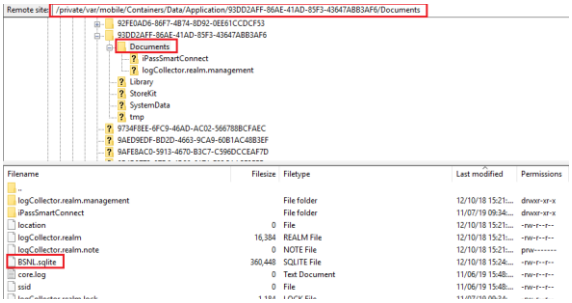


Figure 35: BSNL Path

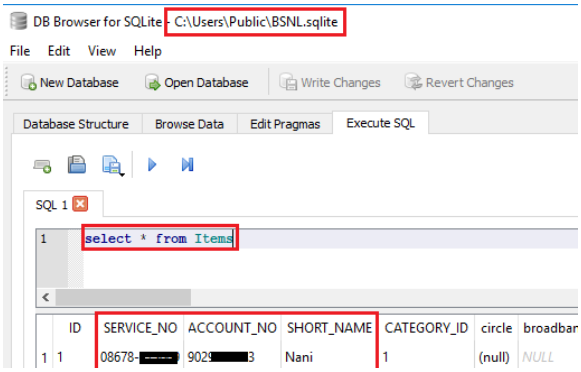


Figure 36: Extracted data

WYNK: This Music App is used to find the list of MP3 songs downloaded and login details of the user.

Path:
/private/var/mobile/Containers/Data/Application/9AFE8AC0-5913-4670-B3C7-C596DCCEAF7D/Documents/stream

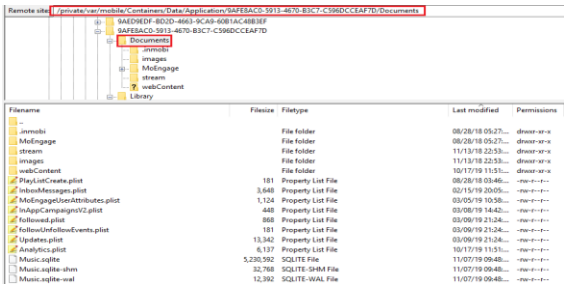


Figure 37: WYNK Path

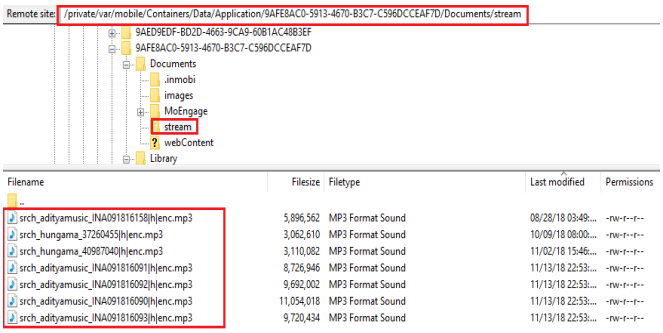


Figure 38: The Downloaded MP3 files

VII. CONCLUSION

Smart phones usage has been expanded vastly with the increase in advanced features like face lock, fingerprint unlock, gaming, and storage capacity etc. Among them iPhone usage is in high growth due to its security policies and features like device encryption.

So, extracting data from iPhone is very challenging in Forensics. By using the method proposed in this paper, the artifacts can be analysed which helps the forensic investigator to extract the data and prepare an evidence.

ACKNOWLEDGEMENTS

This work is assisted by my project guide from Department of Science and Technology and ESF Labs. I would be very grateful to my project guide and the individuals in the company who helped me to complete my project successfully at each and every phase of execution.

REFERENCES

1. Ahmad Raza Cheema, Mian Muhammad Waseem Iqbal, Waqas Ali "Analysing iPhone File System files with Open Source Toolkit for Forensic Evidence" – Research Gate
2. Feng Liu, Ke-sheng Liu, Chao Chang Wang, Yan Wang "Research on the technology of iOS jailbreak" - IEEE
3. Aswami Ariffin, Christian D' Orazio, Kim-Kwang Raymond Choo, Jill Slay "iOS Forensics: How can we recover deleted images files with timestamp in a forensically sound manner" - IEEE
4. Ryan R Kubasiak, Sean Morrissey, Walter Barr, James Kelly Brown, Max Caceres, Mike Chasman and James Cornell "Mac OSX, iPod, and iPhone Forensic Analysis"
5. Adam Shortall, M A Hannan Bin Azhat "Forensic Acquisitions of WhatsApp data on popular mobile platforms"-IEEE
6. M. Bader and I.Baggili, "iPhone 3GS Forensics: logical analysis using Apple iTunes backup utility" Small Scale Digital Forensics
7. Christian D' Orazio, Aswami Ariffin, Kim-Kwang Raymond Choo "iOS Anti-Forensics: How can we security conceal, delete and insert data?" -IEEE
8. "Mobile Device Forensics", Wikipedia
9. "Jailbreaking", Wikipedia
10. "iOS Application Structure", https://developer.apple.com/library/archive/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemOverview/FileSystemOverview.html
11. S. Brothers, "Cell phone and GPS Forensic Tool Classification System" – Presentation to Digital Forensics
12. Mona Erafani Joorabchi, Ali Mesbah "Reverse Engineering iOS Mobile Applications"- IEEE
13. Mohd Shahdi Ahmad, Nur Emyra Musa, Rathidevi Nadarajah, Rosilah Hassan, Nor Effendy Othman "Comparison Between Android and iOS Operating System in terms of security" - IEEE
14. Wencheng Yan, Jiankun Hu, Clinton Fernandes, Vijay Sivaraman, Qianhong Wu "Vulnerability Analysis of iPhone 6" - IEEE
15. Roy Want "iPhone: Smarter Than the Average Phone" - IEEE
16. Anusha, A., Guptha, A., Sivanageswar Rao, G. & Tenali, R.K. 2019, "A model for smart agriculture using IOT", International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 6, pp. 1656-1659.
17. Daya Sagar, K.V., Siva Nageswara Rao, G., Srikanth, T. & Raghavendra, K. 2014, "A relational analytic platform with Hadoop using the On Demand Integration (ODI) capability", International Journal of Applied Engineering Research, vol. 9, no. 13, pp. 2095-2102.
18. Phani Babu, K. & Siva Nageswara Rao, G. 2019, "Smart healthcare system", International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 12, pp. 4184-4188.
19. Siva Nageswara Rao, G. 2016, "Real time scheduling for dynamic process execution using round robin switching", Journal of Theoretical and Applied Information Technology, vol. 89, no. 1, pp. 60-66.
20. Siva Nageswara Rao, G., Jayanth, T.V.M., Shabzan, S. & Supritha, B. 2017, "Airport luggage tracking system using RFID and GSM", Journal of Advanced Research in Dynamical and Control Systems, vol. 9, pp. 748-757.
21. Siva Nageswara Rao, G., Kiran Babu, M. & Rao, S. 2019, "Efficient scheduling measures for improving real time systems using prediction process", Journal of Computational and Theoretical Nanoscience, vol. 16, no. 5-6, pp. 1876-1880.
22. Siva Nageswara Rao, G., Krishna, C.V.P. & Rao, K.R. 2014, "Extreme Programming for service-based application development architecture", Proceedings of the 2014 Conference on IT in Business, Industry and Government: An International Conference by CSI on Big Data, CSIBIG 2014.
23. Siva Nageswara Rao, G., Krishna, C.V.P. & Rao, K.R. 2014, Multi Objective Particle Swarm Optimization for Software Cost Estimation.
24. Siva Nageswara Rao, G., Krishna, C.V.P. & Rao, K.R. 2013, "Rational unified process for service oriented application in extreme programming", 2013 4th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2013.
25. Siva Nageswara Rao, G., Manojkumar, B., Jaya Raj, R. & Sharma, A. 2018, "IOT based garbage management system", Journal of Advanced Research in Dynamical and Control Systems, vol. 10, no. 4, pp. 31-36.
26. Siva Nageswara Rao, G., Srinivasu, N., Girish Kumar, K. & Abhishek, B. 2014, "An enhanced dynamic Round Robin CPU scheduling algorithm", International Journal of Applied Engineering Research, vol. 9, no. 15, pp. 3085-3098.
27. Siva Nageswara Rao, G., Srinivasu, N. & Rama Koteswara Rao, G. 2015, "Dynamic time slice calculation for round robin process scheduling using NOC", International Journal of Electrical and Computer Engineering, vol. 5, no. 6, pp. 1480-1485.
28. Siva Nageswara Rao, G., Srinivasu, N., Sagar, K.V.D. & Sai Madhuri, P. 2014, "Comparison of round robin CPU scheduling algorithm with various dynamic time quantum", International Journal of Applied Engineering Research, vol. 9, no. 18, pp. 4905-4916.
29. Siva Nageswara Rao, G. & Srinivasu, S. 2016, "Task scheduling for real time applications using mean-difference round robin (MDRR) algorithm with dynamic time slice (MDDRWDTS)", International Journal of Pharmacy and Technology, vol. 8, no. 3, pp. 16082-16088.
30. Siva Nageswara Rao, G. & Srinivasu, S.V.N. 2017, "Hybrid approach for task scheduling in heterogeneous cloud based systems", Journal of Advanced Research in Dynamical and Control Systems, vol. 9, no. Special Issue 14, pp. 618-625.
31. Siva Nageswara Rao, G. & Srinivasu, S.V.N. 2016, "An efficient round robin cpuscheduling algorithm using dynamic time slice", International Journal of Pharmacy and Technology, vol. 8, no. 4, pp. 21461-21469.
32. Siva Nageswara Rao, G., Srinivasu, S.V.N., Srinivasu, N. & Naga Raju, O. 2015, "A new proposed Dynamic dual processor based CPU scheduling algorithm", Journal of Theoretical and Applied Information Technology, vol. 73, no. 2, pp. 226-231.
33. Siva Nageswara Rao, G., Srinivasu, S.V.N., Srinivasu, N. & Ramakoteswara Rao, G. 2015, "Enhanced precedence scheduling algorithm with dynamic time quantum (EPSADTQ)", Research Journal of Applied Sciences, Engineering and Technology, vol. 10, no. 8, pp. 938-941.
34. Srilakshmi, M., VenkataKrishna, S. & Siva Nageswara Rao, G. 2014, "Dynamically managing the software components in deployment based architecture", International Journal of Applied Engineering Research, vol. 9, no. 19, pp. 6139-6148.
35. Uppuluri, S., Dilip Raja, K., Sivaleela, D., Suresh Reddy, A. & Siva Nageswara Rao, G. 2015, "Effect of ethanolic pod extract of Canavalia gladiata on peptic ulcer in wistar rats", International Journal of Pharmaceutical and Clinical Research, vol. 7, no. 6, pp. 383-385.
36. Venkateswara Rao, C. & Siva Nageswara Rao, G. 2019, "An effective research on data mining techniques for intrusion detection & learning classes", International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 11 Special Issue, pp. 845-849.
37. Yaswanth, K.B., Manasa, L., Chetan, B.V. & Siva Nageswara Rao, G. 2019, "Modern vehicle tracking and monitoring system using embedded technology", International Journal of Recent Technology and Engineering, vol. 8, no. 1, pp. 1849-1851.

AUTHORS PROFILE

Hyndavi Koganti is a student pursuing her M.Tech in the field of Cyber Security and Digital Forensics - Department of CSE, KLEF, Vaddeswaram, A.P, India. She is a graduate who completed her B.Tech in computer science and engineering at Amrita Sai institute of Engineering and Technology.

Dr. G Siva Nageswara Rao is a professor in CSE department, KLEF, Vaddeswaram, Guntur. He has totally 23 years of experience in teaching and published 21 scopus, 11 non scopus papers. His reswarch area at present is on task scheduling and software engineering.