NFSU JOURNAL OF CYBER SECURITY & DIGITAL FORENSICS

CSDF
Volume: 1, Issue: 1 July 2022
E-ISSN: In Process

Forensic Investigation: Apple Devices Acquisition & Analysis

Dr. Parag Shukla,
Assistant Professor,
School of Cyber Security and Digital Forensics,
National Forensic Sciences University,
Gandhinagar, Gujarat, India
parag.shukla@nfsu.ac.in

Aditya Pratap
Student, M.Sc. Digital Forensics & Information Security,
School of Cyber Security and Digital Forensics,
National Forensic Sciences University,
Gandhinagar, Gujarat, India
aditya.pratap9557@gmail.com

Abstract— In the field of forensic investigation, Apple devices have always been of great interest. As a result of the device's security features, its architecture as well as its new hardware and software features, investigators have been unable to do much with it. There is limited exposure of Apple devices to investigators since they are more familiar with Android & Windows devices. As part of this research paper, we will present a brief overview of iOS and Mac devices. Understanding the Internal Filesystem, Application Data Storage, Boot Modes & what are the new Security features that even encrypt that data stored within the eMMC. We'll also look into what are the different acquisition and analysis methods available for Apple devices which can be performed via Open-Source tools.

Keywords— Apple Devices, Acquisition, Jailbreak, iOS, Mac, forensic investigation

I. INTRODUCTION

With increase in number of Cyber crimes, the need for forensic investigation comes into play to present the investigation in court of law. And in the recent years, with launch of new and cheap mobile phones as well as more usage of mobile phones for data storage and day to day tasks, have concern over investigating mobile devices.

In investigation of mobile devices, Apple iOS devices are a big concern due to increased security features of the device. When investigating iOS devices, sometimes it involves investigating other Apple device like Macbook.

A. AIM

The project discuss about acquisition and analysis methods to perform on Apple devices including iOS and MacOS with use of open-source tools.

Main objective of the project is to provide awareness about the internals of Apple devices and allow audience to perform the mentioned steps on their own device. The project discuss about the security features, application data, acquisition and analysis methods on devices like iPhone 6 Plus, iPhone 11 Pro Max, iPad 2 WiFi and Macbook Air (M1 Chip).

II. METHODOLOGY

A. IOS Architecture

- Core OS Layer
  - First layer on device hardware
  - Provides low level services like networking, memory management
  - Helps to create and manage certificates and called upon keychain services
- Core Services
  - Provides applications the fundamental services
  - Support for framework like Address, Cloud, Coredata, location etc
- Media Layer
  - Enables audio, video graphics of devices
  - Use frameworks running different libraries to enable the technology
- Cocoa Touch
  - Infrastructure to implement visual interface to apps
  - Support for touch and motion event

B. IOS Boot Process

- Boot ROM
  - Read-only block contains Root Certificate which verifies signature and decrypt Low Level Bootloader (LLB)
- Low Level Bootloader (LLB)
  - LLB contains code invoked by Boot ROM
  - Verifying authority of iBoot and executes it
- iBoot
  - Verify signature of kernel before execution
  - Failure to load iBoot results in DFU or Recovery mode
- Kernel
  - Verify device iOS version and required services and applications

C. IOS Operating Modes

- Normal Mode

- By-default mode to allow user access apps and data from interface
- Device Firmware Upgrade (DFU) Mode
  - Used for upgrading or downgrading iOS versions
  - Can be used to perform Physical acquisition
- Recovery Mode
  - Bypass loading of OS by booting in Stage 2 bootloader
  - After device turned off, it cannot complete a boot cycle without help of computer based jailbreak application and physical connection between device and computer
  - Example, redsn0w (for iOS devices with A4 chip)
- Semi-Tethered Jailbreak
  - Permits handset to complete boot-cycle after being pawned
  - But jailbreak extensions won't load until computer based application is deployed over physical cable connection between device and computer
  - Example, checkra1n (for iOS devices with A7-A11 chip)
- Semi-Untethered Jailbreak
  - Permits handset to complete boot cycle but jailbreak extensions won't load until side loaded jailbreak app on device is deployed
  - Example. Chimera, unc0ver
- Untethered Jailbreak
  - Permits handset to complete boot-cycle after being pawned without any interruptions to jailbreak oriented functionality
  - Example, Pangu, JailbreakMe

E. IOS Acquisition Parameters

- iDevice Model
  - Earlier iOS device models allows easy file system acquisition
- iOS Version
  - Acquisition is highly dependent on iOS version due to encryption and updated security features
- Passcode
  - User passcode required at time of acquisition
- Backup Passcode
  - Optional feature to create passcode while creating backup
- Jailbroken Device
  - Jailbreak allows easy acquisition and bypassing restrictions

F. IOS Lockdown Certificate

- Lockdown certificate created on system when device connected for first time with iTunes
  - Can be used to perform activation of device as well as for Logical acquistion

D. IOS Jailbreaking

Jailbreaking allows to remove the barriers set by manufacturer manually is Jailbreak; this unlocking process is possible with special software that modifies iOS

- Tethered Jailbreak
  - Temporarily pawns handset for single boot-cycle
- It stores the UDID data for iOS devices that are synced using iTunes
- Once this certificate is generated, no longer is required to unlock the device when connected to same device again
- This can be used to gain partial access to the device without knowing the passcode of the iOS device

III. IMPLEMENTATION

A. Environment Setup

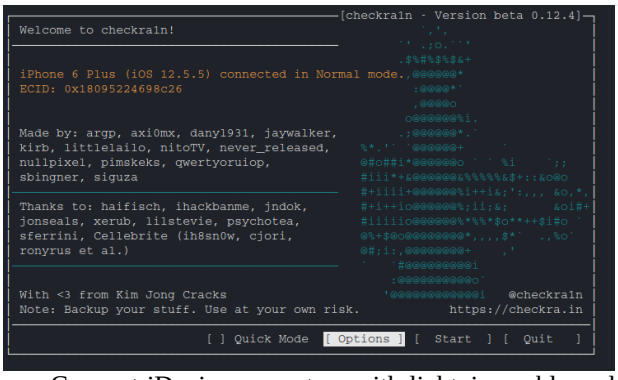
- 3uTools
  - All in one tool for iOS Devices
  - Allows to take backup, jailbreak & manage apps, photos and other multimedia files
  - Full view of iOS device status including activation, battery and iCloud lock status as well as detailed iOS & iDevice information
  - Additional feature of flashing & downgrading firmware
- Checkra1n
  - Community project which provides semi-tethered jailbreak that are based on "checkm8" exploit
  - Exploitation supports A10 and A10X chipsets
- iLEAPP
  - Developed by Alexis Brignoni for analysis of physical image of iOS device
  - Provide detailed report after analysis process
- Belkasoft Evidence Center X
  - A creation of Belkasoft for the forensic analysis of computer, mobile and cloud platforms
  - Helps to acquire and analyse wide range of mobile devices and creating report as well
- Sumuri Recon Lab
  - Forensic suite developed by Sumuri specially for Apple devices
  - Allows to take Windows, Mac, iOS, Android and Google takeout automated analysis

B. Jailbreaking iOS

- On Kali Linux terminal, install checkra1n by command sudo apt-get install checkra1n

NFSU JOURNAL OF CYBER SECURITY & DIGITAL FORENSICS

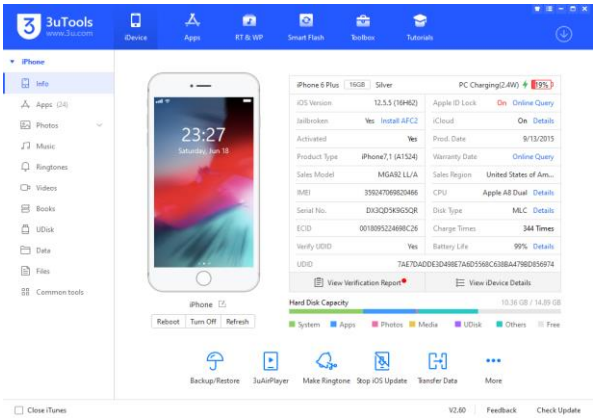
CSDF
Volume: 1, Issue: 1 July 2022
E-ISSN: In Process

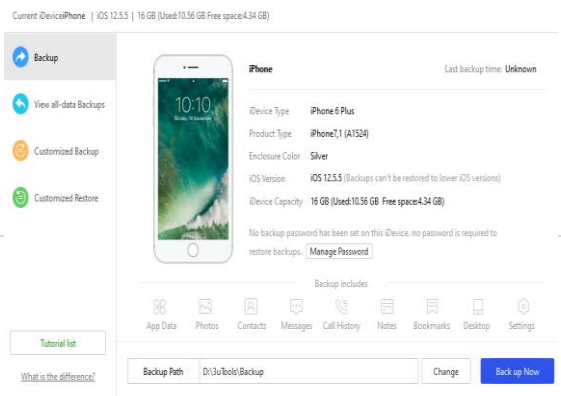


- Connect iDevice on system with lightning cable and run checkra1n with sudo privileges
- Select start and iDevice will be put in DFU mode before starting with checkm8 exploit
- After iDevice booted to normal mode, checkra1n app visible on springboard
- Open the checkra1n app and select option “Install Cydia” to finish the jailbreak process

C. *iOS Logical Acquisition with 3uTools*

- On Windows, connect the iDevice and open 3uTools



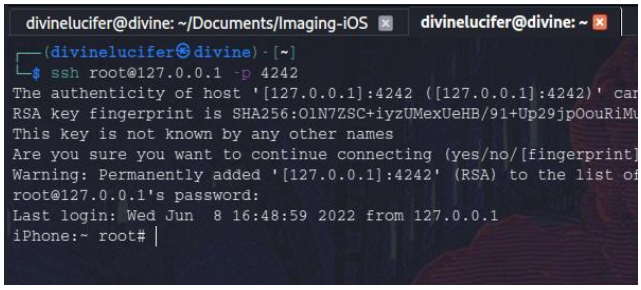


- Select “Backup/Restore” option at bottom

D. *iOS File System Imaging with Linux Terminal*

- On Linux, connect iDevice and within terminal; *iproxy 4242 22*

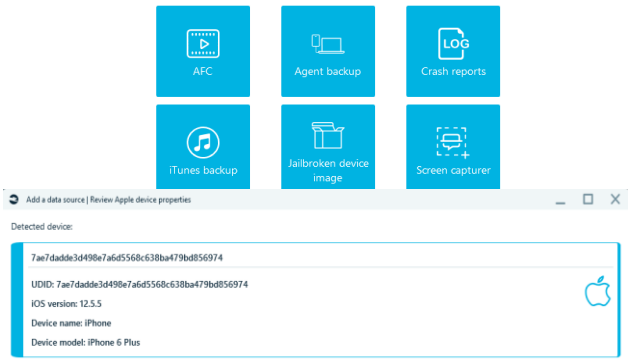
- Open second terminal and write command; *ssh root@127.0.0.1 -p 4242*



- Identify partition of iPhone using command; *df -H*
- Within the system terminal, type the command to create tar image of the iPhone partition; *ssh root@127.0.0.1 tar czf – private/var > iPhone_var.tar*

E. *iOS Physical Acquisition with Belkasoft Evidence Center*

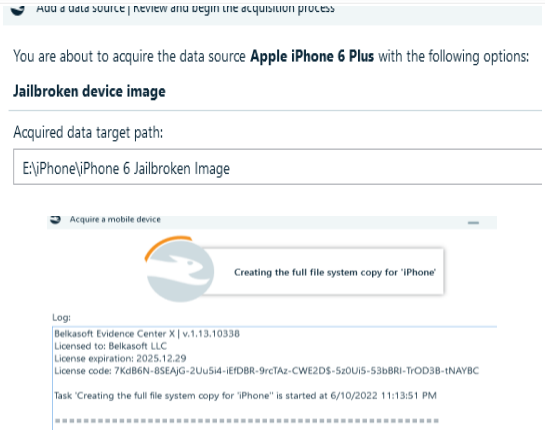
- Open Belkasoft Evidence Centre X and connect iDevice to the system
- In “Add data source”, select “Acquire > Mobile Image”
- After selecting model of iDevice, choose “Jailbroken device Image” and Belkasoft will display the detected device



- Enter the path to save the iDevice image and Belkasoft will start process

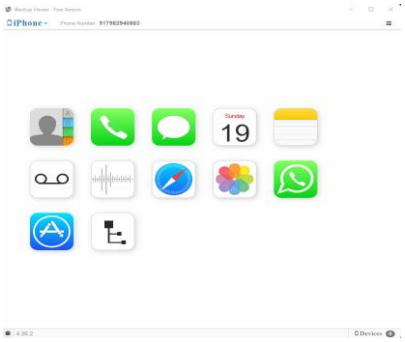
NFSU JOURNAL OF CYBER SECURITY & DIGITAL FORENSICS

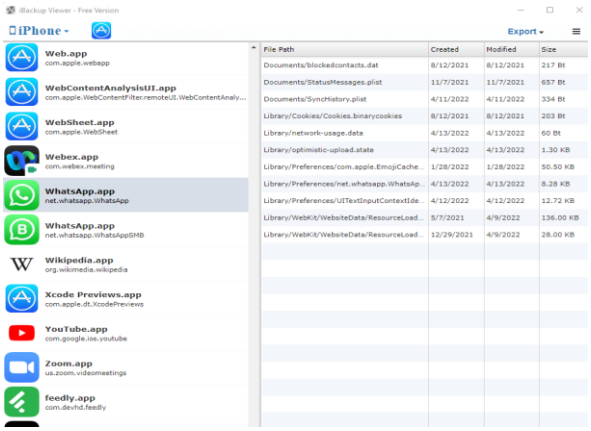
CSDF
Volume: 1, Issue: 1 July 2022
E-ISSN: In Process



F. iOS Backup/Logical Image Analysis with iBackup viewer

- Open iBackup Viewer and browse the iOS backup and select "manifest.db" file
- Home page will display the default applications installed within the device. For eg, view the whatsapp chats selecting the "WhatsApp" icon
- Select "App Store" icon to view all installed applications on the iDevice and view the respected application data stored within

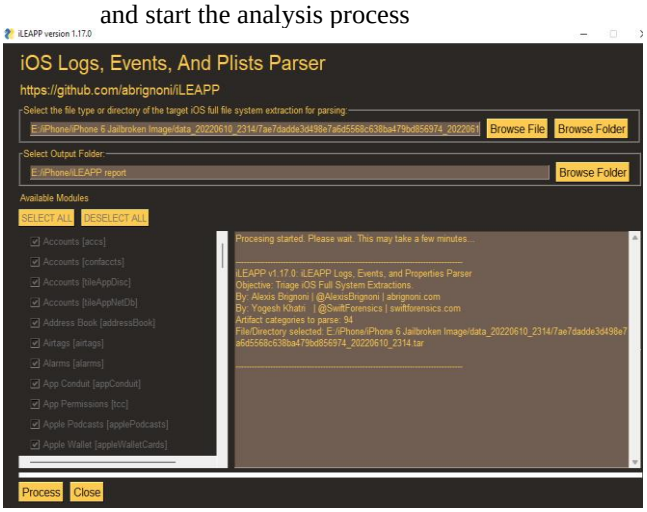




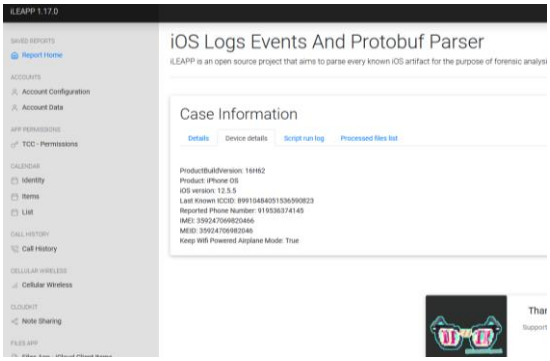
- iBackup viewer allows to export any selected data and examine the files with applications like SQLite DB Browser, Notepad++, 3uTools, HexEditor

G. iOS Filesystem Image Analysis with iLEAPP

- Open GUI version of iLEAPP and select the filesystem image file path
- Select all the modules to analyze within the image and start the analysis process



- After the analysis process is completed, iLEAPP creates HTML report for the process
- Analysis provides different sections to view such as "Account data, Call History, WhatsApp messages, WiFi known networks etc."



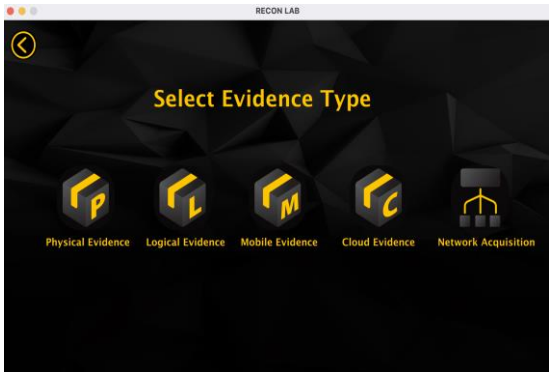
H. Mac OS Triaging with Sumuri Recon Lab

Sumuri Recon Lab is all in one tool for triaging, imaging, acquiring and analyzing the apple devices. The tool also has additional support for imaging Mac devices including the M1 and M2 chipsets.

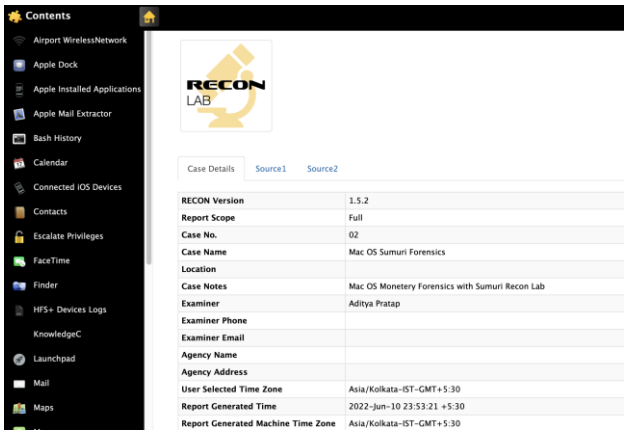
- Install Sumuri Recon Lab on Mac and run the application with proper privilege permissions
- Select "Logical Evidence" from Evidence Type that will display the logical volumes present inside the Mac device

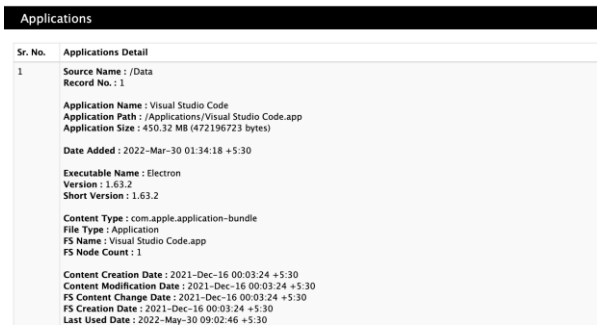
NFSU JOURNAL OF CYBER SECURITY & DIGITAL FORENSICS

CSDF
Volume: 1, Issue: 1 July 2022
E-ISSN: In Process



- Once the analysis procedure is completed, provides a list of artifacts including
  - Apple Installed Applications
  - Bash History





  - Escalate Privileges
  - Connected iOS devices

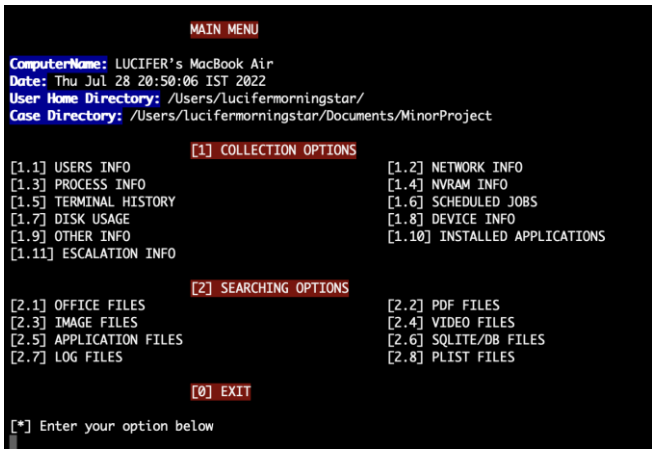
I. Mac OS Triaging via Mac Terminal

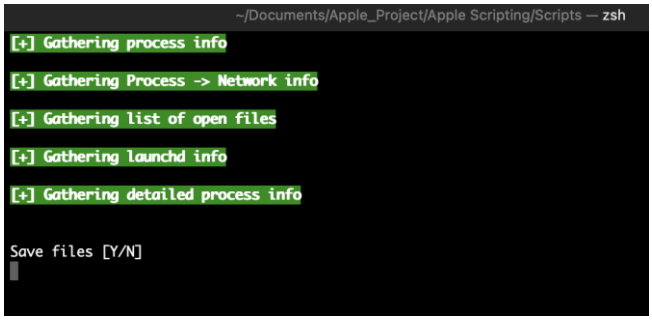
Terminal allows to gather acquire the volatile information as well as files information stored within the Mac device.
For this purpose, script is developed for Mac OS Monterey version 12.3 to allow investigator to perform the most basic operations such as:

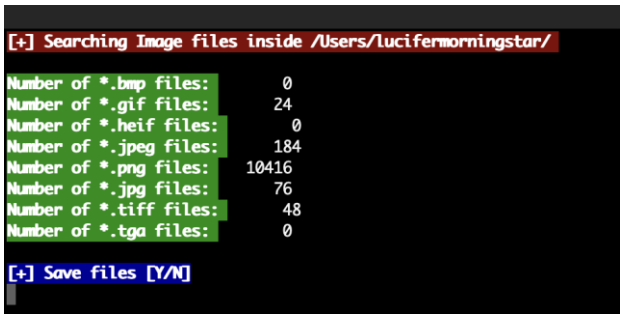
- Collect user info, process info, network information
- Determine disk usage, installed applications etc
- List out different office files, application files, pdf files, multimedia files etc
- All the output can be saved dynamically by the investigators

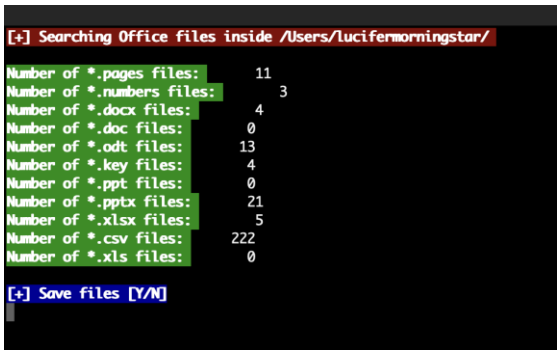
NFSU JOURNAL OF CYBER SECURITY & DIGITAL FORENSICS

CSDF
Volume: 1, Issue: 1 July 2022
E-ISSN: In Process









V. RESULTS & DISCUSSIONS

There are some limitations when analyzing with commercial tools like Sumuri Recon Lab and Belkasoft Evidence Center X. There are some features that commercial tools lack if the focus of analysis is application-based, namely extracting data from installed applications

It is quite useful to have open-source tools available for Apple devices with an understanding of iOS and Mac file systems

Jailbreaking iOS devices allow to obtain privileged access on the iDevice and allows to extract physical image via SSH. Free tool like 3uTools allow to take iOS backup (Logical Image) of the device and analyze the same with the tool.

From the analysis conducted on the different devices, common artifacts that can be found are as follows:

Common IOS Artifacts Location

IOS ARTIFACTS LOCATION	
DESCRIPTION	**PATH**
DEVICE INFORMATION	
Operating System	/private/var/installd/Library/MobileInstallation/ LastBuildInfo.plist
Last BootTime	/private/var/mobile/Library/Preferences/ com.apple.aggregated.plist
IMSIs	/private/var/mobile/Library/Preferences/ com.apple.mmcs.plist
Device name	/private/var/mobile/Library/Preferences/ com.apple.mobilegestalt.plist
PASSWORDS AND ACCOUNT INFORMATION	
Account information	/private/var/mobile/Library/Accounts/Accounts3.sqlite
Phone number	/private/var/mobile/Library/Preferences/ com.apple.commcenter.shared.plist
APPLICATION USAGE	
Mobile Applications Installation Logs	/private/var/installd/Library/Logs/MobileInstallation/ mobile_installation.log.
Application traces	/private/var/mobile/Library/AggregatedDictionary/ ADDataStore.sqlitedb
Installed Apps and Apps Path	/private/var/mobile/Library/AppConduit/ AvailableApps.plist
LOCATION ARTIFACTS	
Seen Bluetooth devices	/private/var/containers/Shared/SystemGroup/<GUID>/ Library/Database/ com.apple.MobileBluetooth.ledevices.other.db
Apple Maps history	/private/var/mobile/Containers/Data/Application/ <Apple_Maps_GUID>/Library/Maps/ GeoHistory.mapsdata
Last latitude and longitude, map search history	/private/var/mobile/Containers/Data/Application/ <Apple_Maps_GUID>/Library/Preferences/ com.apple.Maps.plist
Apple Maps bookmarks	/private/var/mobile/Containers/Data/Application/ <Apple_Maps_GUID>/Library/SyncedPreferences/

IOS ARTIFACTS LOCATION	
DESCRIPTION	**PATH**
NETWORK CONNECTIONS	
Network data usage per App	/private/var/networkd/netusage.sqlite
Network Extension	/private/var/preferences/com.apple.networkextension.plist
Network IP, Wi-Fi, Cellular	/private/var/preferences/SystemConfiguration/ com.apple.networkidentification.plist
Wi-Fi	/private/var/preferences/SystemConfiguration/ com.apple.wifi.plist
Wi-Fi Mac Addresses	/private/var/preferences/SystemConfiguration/ NetworkInterfaces.plist
MULTIMEDIA ARTIFACTS	
Photos	/private/var/mobile/Library/Preferences/ com.apple.mobileslideshow.plist
MMS File	/private/var/mobile/Library/SMS/Attachments/
User Created/Saved Photos	/private/var/mobile/Media/DCIM/1*APPLE
iTunes Media Library	/private/var/mobile/Media/iTunes_Control/iTunes/ MediaLibrary.sqlitedb
BROWSER ACTIVITY	
Safari Cache files	private/var/mobile/Containers/Data/Application/<Apple Safari GUID/Library/Caches/com.apple.mobilesafari/
Safari Cache database	/private/var/mobile/Containers/Data/Application/<Apple Safari GUID/Library/Caches/com.apple.mobilesafari/Cache.db
Safari Website cache	/private/var/mobile/Containers/Data/Application/<Apple Safari GUID/Library/Caches/com.apple.WebAppCache/ApplicationCache.db
Safari Cookies	/private/var/mobile/Containers/Data/Application/<Apple Safari GUID/Library/Cookies/Cookies.binarycookies

iOS Physical image open-source analysis tool developed by Alexis Brignoni "iLEAPP", create detailed report for the iDevice with information like device details, call info, contacts info, messages info, applications info etc.

Commercial tool like Belkasoft Evidence Center & Sumuri Recon Lab provides the feature for Timeline analysis, Geo-location analysis, Link analysis are among some of the advanced features

Newly launched iOS applications may not be analyzed in detail by commercial tools

Common Mac OS Artifacts Location

MAC OS ARTIFACTS LOCATION	
DESCRIPTION	**PATH**
Recent Items	/Users/%user%/Library/Application Support/com.apple.sharedfilelist/com.apple.LSSharedFileList.ApplicationRecentDocuments/ com.apple.textedit.sfl2
Recent Applications	Users/%user%/Library/Application Support/com.apple.sharedfilelist/com.apple.LSSharedFileList.RecentApplications.sfl2
Last Logout Session	/Users/%user%/Library/Preferences/ com.apple.loginwindow.plist
Dock Items	/Users/%user%/Library/Preferences/ com.apple.dock.plist
Installed Applications	/Applications
Mail	/Users/%user%Library/Mail/
Bash History	/Users/%user%/.zsh_history
Connected Devices	/Users/%user%/Library/Preferences/ com.apple.iPod.plist
Contacts	/Users/%user%/Library/Application Support/AddressBook/
Escalate Privileges	/Users/%user%/.zsh_history
FaceTime Account	/Users/%user%/Library/Preferences/ com.apple.imservice.ids.FaceTime.plist
Finder	/Users/%user%/Library/Preferences/ com.apple.finder.plist
Device Logs	/private/var/db/diagnostics/ /private/var/db/uuidtext/
LaunchPad	/private/var/folders/kt/ _dtxp9y52l37k_nt5xkg0v_80000gn/0/ com.apple.dock.launchpad/db/db
Maps	/Users/%user%/Library/Containers/com.apple.Maps/ Data/Library/Preferences/com.apple.Maps.plist
Message Account	/Users/%user%/Library/Preferences/ByHost/ com.apple.imservice.SMS.<>.plist

IV. FUTURE SCOPE OF WORK

While open-source tools and terminal applications can perform all acquisition and analysis steps, some enhancements can be made related to automating these processes.

Automating the acquisition tasks by identifying the iDevice version, jailbreak status, and if not jailbroken then allow to jailbreak the iDevice with appropriate jailbreak tool and perform suitable acquisition method will allow to cut the time taken for the acquisition purpose.

With the aid of Artificial Intelligence, develop new methods for identifying suitable and efficient data from a large database using keyword-based searching

With enhancement in security features, quite difficult to acquire physical image of Mac devices due to integration of

storage device with the motherboard. Therefore, research new method to acquire physical image from Mac devices to recover deleted data from the device

Reversing and analyzing iOS applications to get detailed information that may also cater the need for malware analysis as well would be included in the future advancement.

ACKNOWLEDGMENT

I would like to express my sincere gratitude to my mentor Dr. Parag C. Shukla sir for his continuous support and motivation that enabled to diversify the road-map for the project and providing the golden opportunity to do this wonderful project on the topic Apple Devices Acquisition & Analysis, which also helped me in doing a lot of Research and I came to know about so many new things related to Apple hardware security, application security, jailbreaking etc.

Secondly, I would also like to thank my parents and friends who helped me a lot in finishing this project within the limited time.

REFERENCES

[1] Jaron Bradley,OS X Incident Response, Syngress, 2016, ISBN 9780128044568

[2] Gianluca Tiepolo, iOS Forensics for Investigators, Packt, 2022, ISBN 9781803234083

[3] Jesse Varsalone, Mac OS X, iPod, and iPhone Forensic Analysis DVD Toolkit, Syngress, 2008, ISBN 9780080949185

[4] Kunal Relan, iOS Penetration Testing, Apress Berkeley, 2016, ISBN 9781484223550

[5] Naveen, 'iOS Architecture', August 30 2021, https://intellipaat.com/blog/tutorial/ios-tutorial/ios-architecture/

[6] Apple Inc., Apple Platform Security, May 17 2021, https://support.apple.com/en-in/guide/security/sec59b0b31ff/web

[7] File System Programming Guide, Apple Inc., https://developer.apple.com/library/archive/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemOverview/FileSystemOverview.html#//apple_ref/doc/uid/TP40010672-CH2-SW13

[8] Apple Developer Documentation, Keychain Services, https://developer.apple.com/documentation/security/keychain_services

[9] TheTechPortal, Phoenix Jailbreak Tutorial for iOS 9, https://thetechportal.com/phoenix-jailbreak-tutorial-for-ios-9/

[10] Jack Farley, Forensic Analysis of iTunes Backup, April 14 2019, https://farleyforensics.com/2019/04/14/forensic-analysis-of-itunes-backups/

[11] SANS Institute, DFIR Advanced Smartphone Forensics, https://www.sans.org/posters/dfir-advanced-smartphone-forensics/

[12] Sumuri, Sumuri Recon Lab, https://sumuri.com/software/recon-lab/

[13] Belkasoft, Belkasoft Evidence Center X, https://belkasoft.com/x

[14] iBackupViewer, iMacTools, https://www.imactools.com/iphonebackupviewer/

[15] Alexis Brignoni, iLEAPP, https://github.com/abrignoni/iLEAPP

[16] 3uTools, 3uTools, https://www.3u.com/