

# **A Forensic Analysis of Video Streaming Activities on Android Applications**



By

**Adil Ahmad**

Registration Number: 329214

MSIS-2K20

Supervisor

**Dr. Mehdi Hussain**

DEPARTMENT OF COMPUTING

Thesis submission in fulfillment of the requirements for the degree of  
Master of Science in Information Security (MSIS)

At

School of Electrical Engineering and Computer Science (SEECS),  
National University of Sciences and Technology (NUST),

Islamabad, Pakistan

(February 2022)

## **THESIS ACCEPTANCE CERTIFICATE**

Certified that final copy of MS/MPhil thesis entitled "A Forensic Analysis of Video Streaming Activities on Android Applications" written by Adil Ahmad, (Registration No 00000329214), of SECS has been vetted by the undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: \_\_\_\_\_ 

Name of Advisor: Dr. Mehdi Hussain

Date: \_\_\_\_\_ **26-Jan-2022**

Signature (HOD): \_\_\_\_\_

Date: \_\_\_\_\_

Signature (Dean/Principal): \_\_\_\_\_

Date: \_\_\_\_\_

## Approval

It is certified that the contents and form of the thesis entitled "A Forensic Analysis of Video Streaming Activities on Android Applications" submitted by Adil Ahmad have been found satisfactory for the requirement of the degree

Advisor : Dr. Mehdi Hussain

Signature:  \_\_\_\_\_

Date: 26-Jan-2022

Committee Member 1: Prof. Hasan Ali Khattak

Signature:  \_\_\_\_\_

**26-Jan-2022**

Committee Member 2: Dr. Dr Hasan Tahir

Signature:  \_\_\_\_\_

Date: 27-Jan-2022

Signature: \_\_\_\_\_

Date: \_\_\_\_\_

## Certificate of Originality

I hereby declare that this submission titled "A Forensic Analysis of Video Streaming Activities on Android Applications" is my own work. To the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECs or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECs or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics, which has been acknowledged. I also verified the originality of contents through plagiarism software.

Student Name: Adil Ahmad

Student Signature: \_\_\_\_\_



### Certificate for Plagiarism

It is certified that PhD/M.Phil/MS Thesis Titled "A Forensic Analysis of Video Streaming Activities on Android Applications" by Adil Ahmad has been examined by us. We undertake the follows:

- a. Thesis has significant new work/knowledge as compared already published or are under consideration to be published elsewhere. No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is placed under quotation marks and duly referenced.
- b. The work presented is original and own work of the author (i.e. there is no plagiarism). No ideas, processes, results or words of others have been presented as Author own work.
- c. There is no fabrication of data or results which have been compiled/analyzed.
- d. There is no falsification by manipulating research materials, equipment or processes, or changing or omitting data or results such that the research is not accurately represented in the research record.
- e. The thesis has been checked using TURNITIN (copy of originality report attached) and found within limits as per HEC plagiarism Policy and instructions issued from time to time.

#### Name & Signature of Supervisor

Dr. Mehdi Hussain

Signature : 

## **Acknowledgement**

I would like to thank my supervisor Dr. Mehdi Hussain for his constant guidance and mentorship through my thesis. It was a great learning experience for me. Also, special thanks to the GEC committee members: Dr. Hasan Tahir and Dr. Hasan Ali Khattak for their valuable feedback and encouragement.

## Table of Contents

<b>List of Figures</b> .....	ix
<b>List of Tables</b> .....	x
<b>Abstract</b> .....	xi
<b>1 Introduction</b> .....	1
<b>1.1 Definitions</b> .....	1
<b>1.1.1 Forensics</b> .....	1
<b>1.1.2 Forensic Science</b> .....	1
<b>1.1.3 Digital Forensics</b> .....	2
<b>1.1.4 Forensics Process</b> .....	2
<b>1.1.5 Mobile Device Forensics</b> .....	2
<b>1.1.6 Video Streaming Applications</b> .....	3
<b>1.2 Motivation</b> .....	3
<b>1.3 Relevance to National Needs</b> .....	4
<b>1.4 Research Focus &amp; Questions</b> .....	4
<b>1.5 Problem Statement</b> .....	4
<b>1.6 Scope of Study</b> .....	5
<b>1.7 Research Challenges</b> .....	5
<b>1.8 Document Structure</b> .....	5
<b>2 Literature Review</b> .....	6
<b>2.1 Forensic Research on Android Applications</b> .....	6
<b>2.2 Evaluation of Tools used for Mobile Device Forensics</b> .....	8
<b>2.3 Challenges of Mobile Device Forensics</b> .....	10
<b>2.4 Summary of Literature Review</b> .....	11
<b>3 Research Methodology</b> .....	12
<b>3.1 Mobile Devices Forensic Methodology</b> .....	12
<b>3.1.1 Preservation</b> .....	12
<b>3.1.2 Acquisition</b> .....	13
<b>3.1.3 Examination &amp; Analysis</b> .....	13
<b>3.1.4 Reporting</b> .....	13
<b>3.2 Experimentation Setup</b> .....	13
<b>3.2.1 Tools &amp; Technologies</b> .....	14
<b>3.2.2 Environment Readiness</b> .....	15

3.2.3	Rooting Process .....	18
3.2.4	Data Acquisition Method.....	19
<b>4</b>	<b>Applications Examination &amp; Analysis .....</b>	<b>21</b>
<b>4.1</b>	<b>Netflix.....</b>	<b>22</b>
4.1.1	Remnants of AI.....	22
4.1.2	Remnants of UL .....	24
4.1.3	Remnants of VV .....	25
4.1.4	Remnants of SV.....	26
4.1.5	Remnants of CL .....	26
4.1.6	Remnants of DV .....	27
4.1.7	Remnants of CP.....	29
4.1.8	Remnants of RV .....	29
4.1.9	Remnants of DP.....	29
4.1.10	Remnants of UA .....	29
4.1.11	Summary of Remnants .....	30
<b>4.2</b>	<b>Amazon Prime Video .....</b>	<b>31</b>
4.2.1	Remnants of AI.....	32
4.2.2	Remnants of UL .....	34
4.2.3	Remnants of VV .....	35
4.2.4	Remnants of SV.....	35
4.2.5	Remnants of CL .....	36
4.2.6	Remnants of DV .....	36
4.2.7	Remnants of CP.....	37
4.2.8	Remnants of RV .....	37
4.2.9	Remnants of DP.....	37
4.2.10	Remnants of UA .....	38
4.2.11	Summary of Remnants .....	38
<b>4.3</b>	<b>iFlix.....</b>	<b>39</b>
4.3.1	Remnants of AI.....	40
4.3.2	Remnants of UL .....	42
4.3.3	Remnants of VV .....	42
4.3.4	Remnants of SV.....	43
4.3.5	Remnants of CL .....	43



4.3.6	Remnants of DV .....	43
4.3.7	Remnants of CP.....	44
4.3.8	Remnants of RV .....	44
4.3.9	Remnants of DP.....	44
4.3.10	Remnants of UA .....	44
4.3.11	Summary of Remnants .....	45
5	Results & Discussions .....	47
5.1	Netflix Analysis Summary.....	49
5.2	Amazon Prime Video Analysis Summary.....	51
5.3	iFlix Analysis Summary.....	52
5.4	Comparison of Available Artifacts.....	53
6	Conclusion and Future Horizons.....	55
6.1	Future Work.....	56
7	References.....	57

## List of Figures

Figure 1-1 Categories of Digital Forensics.....	2
Figure 2-1 Forensics Tools Categorization [20].....	9
Figure 2-2 Data Acquisition through an Intermediate Device [22].....	10
Figure 3-1 NIST's Forensic Model.....	12
Figure 3-2 Proposed Methodology .....	14

## **List of Tables**

Table 2-1 Summary of Literature Review .....	11
Table 4-1 Description of the Codes of Groups of Activities .....	21
Table 4-2 Summary of Netflix Remnants with Location Path .....	31
Table 4-3 Summary of Amazon Prime Video Remnants with Location Path .....	39
Table 4-4 Summary of iFlix Remnants with Location Path .....	46
Table 5-1 Summary of Netflix Artifacts .....	50
Table 5-2 Summary of Amazon Prime Video Artifacts .....	52
Table 5-3 Summary of iFlix Artifacts .....	53
Table 5-4 Comparison of Artifacts .....	54

## Abstract

The past decade has seen a major shift in electronic media, especially in the entertainment sector. Now, more people are consuming content through digital media, which was earlier being viewed via electronic media. This has given rise to online video streaming platforms which allow users to view content at the time of their convenience as opposed to electronic media where the transmission has a set schedule. Initially, these platforms could only be accessed through their web-based applications, but over time mobile applications were also developed for these video streaming platforms. These applications tend to store personal information and also leave behind remnants of the activities performed even after the application has been uninstalled. These remnants need to be examined to verify whether they violate the privacy of the user. In the scenario where malicious actors gain access to the mobile device of a user, it can prove detrimental for the user. Furthermore, they can also aid in investigations where the law-enforcing authorities need to cross-check the alibi of the suspects. A very common use case for this can be a road accident where the law enforcement agencies can prove that the suspect was viewing a video on one of these platforms. This thesis will be examining the android applications of the top three and most popular video streaming platforms which are Netflix, Amazon Prime Video, and iFlix. Amongst the three, Netflix is the most popular and has been present for the longest time. Several video streaming activities will be performed through the above mentioned applications on a rooted android mobile device, after which we will take a physical image for analysis. The goal is to present the artifacts left behind the applications along with the path to their location in a well-documented format that will assist the forensic investigators to get a better understanding of the applications' behavior. It was found that Netflix leaves behind the most detailed artifacts of the user's activities whereas iFlix stores the least amount of artifacts on the mobile device.

# Chapter - 1

## 1 Introduction

The following sections will be covered in this chapter:

- Section 1.1 – Definitions of some of the keywords
- Section 1.2 – Motivation to conduct the research in this domain
- Section 1.3 – Relevance of the research to the national needs
- Section 1.4 – Focus of the research along with the research questions
- Section 1.5 – Problem statement
- Section 1.6 – Scope of this study
- Section 1.7 – Challenges encountered in the research
- Section 1.8 – Structure of the thesis document

### 1.1 Definitions

This section will be defining some of the more commonly used terms in this research document that will be necessary for the reader to get a better understanding of this thesis research.

#### 1.1.1 Forensics

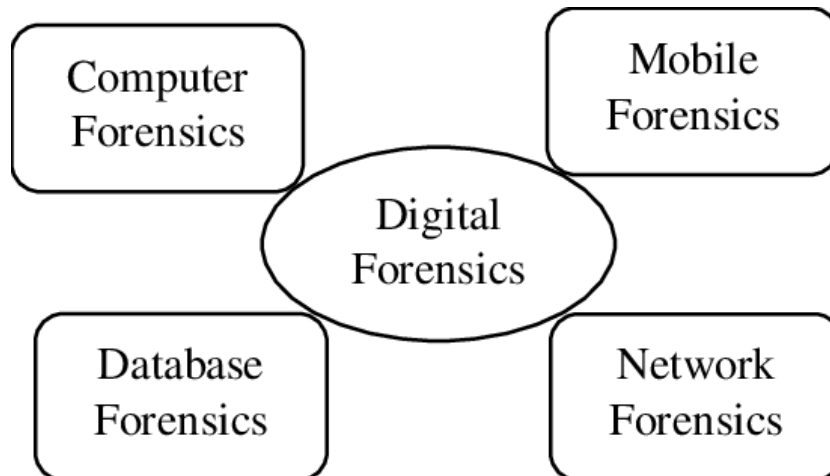
The word forensics is derived from the Latin word “forensic” which means “in open court” or “public”. In the modern age, forensics refers to finding evidence to solve a crime.

#### 1.1.2 Forensic Science

Forensic science is the study involving scientific means to extract and examine the evidence which is to be presented in court. There is a huge range of disciplines where forensic science can be applied. These include biological evidence such as fingerprints and DNA samples, wildlife forensics, and the more recent introduction of digital forensics. All these variants of forensic science have certain things in common. These primarily include the task of extracting and preserving evidence in such a manner that it is admissible in the court of law.

### 1.1.3 Digital Forensics

Digital forensics is the more recent form of forensic science where the focus is on electronic data that can be used as evidence. This electronic data can be present in computers, hard drives, mobile phones, and other storage devices.



*Figure 1-1 Categories of Digital Forensics*

### 1.1.4 Forensics Process

There is a certain methodology applied when it comes to solving a digital forensics investigation. A sequence of steps is usually followed in a particular order. These include collection, examination, analysis, and reporting. During the collection phase, the evidence is identified, extracted, and protected. Next, in the examination phase, the acquired data is processed, whereas in the analysis stage valuable information is extracted from the processed data. Finally, in the reporting stage, the findings are documented.

### 1.1.5 Mobile Device Forensics

An important branch of digital forensics, mobile device forensics deals with evidence extraction in the form of data from mobile devices. Here, special care needs to be taken that the evidence is extracted under forensically sound conditions which can be reproduced.

### **1.1.6 Video Streaming Applications**

Video streaming applications are the mobile applications of video streaming platforms. These platforms and in turn the applications provide the users the opportunity to view entertainment content at the time and place of their choosing. Users can also either stream the content online or can download the content to view it offline later.

## **1.2 Motivation**

This age of digitalization has brought a lot of convenience to the everyday life of individuals. Tasks, which would earlier require manual labor, have now been automated. This transformation has transcended the entertainment sector as well. Traditionally, the only sources of entertainment were going to the cinema or relying on radio and television transmission. However, in the past couple of decades, a new alternative has been introduced in the entertainment sector in the form of video streaming platforms. These online services allow users to view a variety of content at the time and place of their as long as they are connected to the internet. Furthermore, they can also download content that can later be viewed offline.

Along with the convenience offered by video streaming platforms, there are also risks involved with it as well. The mobile applications of the video streaming platforms tend to store a lot of information which can have negative consequences if the mobile device of the user is accessed by malicious actors. There is another angle to it as well, the stored information by the mobile applications can prove helpful for the law enforcement agencies to help solve crimes or any criminal activities such as a car accident where the driver was suspected of using a video streaming application while driving, which is considered illegal. Here, with the help of digital forensics the law enforcement agencies can extract artifacts from the driver's mobile device which can serve as evidence to prove the crime.

Forensic studies have been conducted on various types of applications such as social media and instant messaging applications. However, there is a gap in the literature as no one has conducted a forensic analysis of video streaming applications. This presents us with a necessity to study and analyze these video streaming applications to determine the remnants left behind by them along with the path of the location at which they are stored. This will assist the law enforcement agencies in swift action.

### **1.3 Relevance to National Needs**

Pakistan, being a developing economy, is seeing a huge rise in the number of subscriptions to these video streaming platforms. In order to ensure a hassle-free experience for the users, a detailed forensic study of the video streaming applications is required to analyze the extent to which the personal data of users is stored. This research project aims to study those remnants left behind by the video streaming applications. It will serve as a guiding document for the application developers to further improve the security of their applications.

### **1.4 Research Focus & Questions**

This research project is focused on the latest release versions of the top three most popular android video streaming applications in order to extract the remnants left behind them. The three applications are namely Netflix, Amazon Prime Video, and iFlix. Two very important research questions were kept in mind while conducting the research project:

- What is the nature of the remnants left behind corresponding to various activities by the android video streaming applications?
- Secondly, how do the remnants left behind by the three applications compare to one another?

### **1.5 Problem Statement**

Mobile applications of video streaming platforms store a lot of information on mobile devices which can have both positive and negative impacts. Positive, in the sense that it could assist law enforcement agencies in solving crime, and the negative impact is that it could be accessed by malicious actors. This research project will focus on identifying stored artifacts on the mobile devices left behind by the android video streaming applications. It will give law enforcement agencies and forensic investigators a clear direction when it comes to extracting evidence to solve a crime. On the other hand, it will notify the mobile application developers on how to further improve the security of their mobile applications.

Following are the notable contributions of this research project:

- Identified sensitive artifacts left behind on the mobile device by the target applications
- Establish a timeline of activities of a suspect with regard to the video streaming applications



- Account information of a suspect such as his name and email id

## **1.6 Scope of Study**

Samsung Galaxy Note 5 with 32 GB Storage and 4 GB RAM was used for this study. The operating system on the mobile device was Android Version 7. We chose to study the non-volatile memory, which is the storage of the mobile device, for the analysis. The three major android video streaming applications (Netflix, Amazon Video Prime, and iFlix) were chosen based on their popularity which was gauged by the number of downloads they had on the Google Play Store.

In our literature review, we established that no forensic work had been conducted on similar applications before. We perform physical acquisition as opposed to logical acquisition to acquire maximum data and perform our analysis using open source tools.

## **1.7 Research Challenges**

Android application developers are constantly working to improve the security of their applications which means that the forensics process is getting harder over time. Most of the information now stored on the device is in encrypted format which makes retrieval of original data without the encryption key is a difficult task. Physical acquisition is possible on rooted devices and mobile vendors are making it difficult to root the mobile devices. Furthermore, there are so many applications present on a mobile device and each application records the activities in such detail that it gets difficult to navigate through them and get to the required piece of information in a timely manner becomes a challenge for an investigator. Finally, most of the video streaming applications required credit cards for subscription and without access to a credit card, it will be difficult to fully study all the features of the application.

## **1.8 Document Structure**

The document is structured into the following chapters:

- Chapter 1 deals with the definitions, motivation, problem statement, scope, and research challenges
- Chapter 2 gives an in-depth literature review
- Chapter 3 covers the research methodology and the experimentation setup
- Chapter 4 goes over the actual examination and analysis of android video streaming applications
- Chapter 5 analyzes and documents the results in a concise manner
- Chapter 6 concludes the research and provides direction for future work

## Chapter – 2

### 2 Literature Review

In this chapter, we will be reviewing the existing literature pertaining to the domain of android application forensics. The research appears for the literature review were gathered from credible sources which include Google Scholar, ACM Digital Library, Science Direct, and Research Gate. The research papers were selected based on their relevance to the searched queries containing keywords related to the said domain. The literature review has been divided into the following sections based on their sub-domain:

- Forensic Research on Android Applications
- Evaluation of Tools used for Mobile Device Forensics
- Challenges of Mobile Device Forensics

#### 2.1 Forensic Research on Android Applications

Most of the forensic work conducted on android applications has dealt with instant messaging applications. Over the last decade, we have seen a huge rise in the popularity of instant messaging applications that can be credited to the ubiquity of mobile devices. People share all sorts of personal information through these messaging applications, be it in the form of text or media, and a lot of this personal information gets left behind on the mobile device through which they are accessing the applications. Now let's take a closer look at how researchers have extracted artifacts related to these android applications.

M.A.K. Sudozai et al. [1] carried out a forensic research study on the IMO call and chat app on both Android and iOS-based mobile devices. They examined both the artifacts left on the mobile device as well the network traffic transmitted while the application was running. Special emphasis was laid on extensively analyzing the encrypted network traffic. Furthermore, the file structure of the IMO app was studied and it was established that what information is present in the different folders and file locations. Interesting artifacts were discovered which included audio, video, text, and image messages along with the personal information of the contacts of the user. Another significant discovery was the links of the IMO server at which the content was being uploaded. The researchers were able to access the information at those links without any authentication. On the network analysis side, it was found out that even if all the IMO servers are on the firewall, the application still keeps on working by maintaining connections with google servers on port number 443. J. Gregorio et al. [2] conducted forensic analysis on Telegram Messenger application on Windows mobile devices where the author tried to find artifacts which relate to criminal offenses being conducted using the application. Shawn Knox et al. [3] targeted the Happn social dating app which revealed privacy risks.

Asmara Afzal et al. [4] also performed forensic analysis on an instant messaging application named Signal Messenger App. In their research, they targeted how secure instant messaging applications can be used for conducting crimes. Since the communication is end-to-end encrypted, the criminals can use this fact to their advantage as forensic analysis becomes difficult for such type of communication. The researchers opted for a network forensic strategy to identify artifacts. This was done by examining the payload patterns of the encrypted traffic. Researchers were able to detect activities such as text, audio, video, and image messages as well as calls. The list of chat servers and IP addresses involved was also acquired. Encrypted network traffic was also analysed by Gaofeng He et al. [5] where he used IP addresses and DNS Queries to reach his conclusions. Daniel Walnycky et al. [6] conducted both network and device forensics on android social messaging applications and significant artifacts were revealed.

A similar instant messaging application by the name of Line Messenger was forensically analysed by Ammar Fauzan et al. [8] where they laid emphasis on identifying cybercrime. Whatsapp has a feature called Whatsapp Web where the user can access his chats through a web browser. This was found susceptible to wiretapping by Nuril Anwar et al. [9] in their forensic study. Noora Al Mutawa et al. [10] conducted forensic analysis on major social media applications on Blackberry, Android and iOS device. It was found that Blackberry devices store the minimum amount of artifacts on the mobile device. Fazeel Ali Awan [11] also conducted his forensic research on social networking applications where he reached a similar conclusion. Hao Zhang et al. [12] also conducted a forensic study on social media applications where they established which applications store artifacts in encrypted format and which in unencrypted format. Emails applications were forensically analyzed by Rusydi Umar et al. [7] and digital evidence was acquired.

In another study, Hijrah Nurhairani et al. [13] worked on the android application of the social media platform Twitter. In their study, they examined the differences in the artifacts acquired from a rooted and non-rooted android mobile device after running the Twitter application on it. The motive behind this research was to determine if any criminal activity was taking place on the application such as hate speech, cyberbullying, and stalking. The forensic methodology of the National Institute of Justice was followed which includes 5 stages which are identification, collection, examination, analysis, and reporting. The results showed that more evidence was obtained from the rooted phone as compared to the non-rooted phone. Furthermore, the evidence from the rooted phone proved the existence of hate speech by the user. Twitter website was forensically analysed by Revina Saputra et al. [14] where they found artifacts related to text, images and videos shared.

Internet users today face a lot of privacy issues, which has led to a shift towards private browsers. Muhammad Raheel Arshad et al. [15] conduct a forensic analysis on one such

private browser called TOR which is based on onion routing. The analysis was conducted on Windows 10 and Android 10 devices. The study negates the privacy and anonymity claims made by the TOR Project as the researchers were able to retrieve significant artifacts that revealed details about the user's browsing activities.

## **2.2 Evaluation of Tools used for Mobile Device Forensics**

There is an abundance of forensic tools used for a variety of different purposes. Some of these tools are proprietary while others are open source. In this section, we will be going through the existing literature that evaluates these forensic tools in order to guide the forensic investigators.

Guntur Maulana Zamroni [16] et al. compared the success of various forensic tools while extracting artifacts of instant messaging applications such as WhatsApp. The following forensic tools were used:

- Belkasoft Evidence
- Oxygen Forensic
- Magnet AXIOM
- WA Key/DB Extractor

It was concluded that the combination of Magnet AXIOM and WA Key/DB Extractor produced the best results and helped in acquiring the maximum artifacts. Similar research was conducted by Imam Riadi et al. [17] where the authors evaluated forensic tools for crime investigation in instant messaging applications.

Graeme Horsman [18] in his research paper argued that software tools play a major role in the field of digital forensics as they are heavily relied upon for the acquisition and examination stages of the forensic process. Therefore, these tools must be reliable and the results obtained from these tools should be repeatable, only then will the tools be considered as valid sources of evidence from a legal point of view. This study examines the condition of the current digital forensic tools and the difficulties encountered while testing the tools. The researcher concludes that a lot more work is required in the field of testing the forensic tools and provides solutions that involve either a centralized or federated approach when it comes to testing the tools. Manar Abu Talib, in his research [19] also discussed how most forensic tools are the close source and therefore require a black box testing methodology.

Tinu Wu et al. [20] analyzed over 800 articles that were published over a span of 5 years in different reputable conferences and journals in order to find the digital forensics tools which were proposed and developed in research papers. The authors found 62 tools in

total out of which 33 were publically available. A thorough analysis was performed on these tools which included code reviews and examining the available documentation to check whether these tools were being maintained and kept up to date. The researchers categorized the tools into 7 categories based on the Netherlands Register of Council Experts(NRGD). Here are their results:

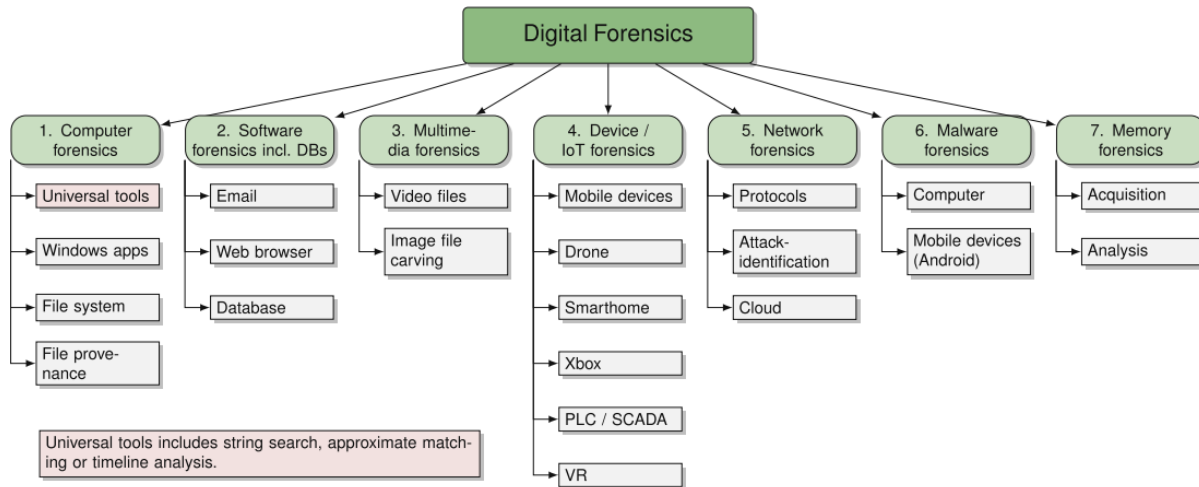


Figure 2-1 Forensics Tools Categorization [20]

In [21], Htar Htar Lwin et al. performed a comparative analysis of android mobile forensics tools for data acquisition and analysis. For the data acquisition stage, the tools that were used were ADB Backup, DD, Belkasoft, and Magnet Acquire. Belkasoft is a commercial tool while the rest are open source. It was found out that ADB Backup can only be used for logical backup, DD can only acquire physical backup and the other two can perform both logical and physical backup. The researchers recommended DD for physical data acquisition and Magnet Acquire for logical data acquisition. For the analysis stage, two tools were compared. The first was Autopsy, which is an open-source tool, and Belkasoft, which is a commercial tool. It was discovered that autopsy has certain limitations like the inability to decrypt passwords in an encrypted format.

With time, acquiring data from android mobile devices is becoming challenging. This is because of frequent updates in the Android operating system and upgrading the security of the devices as well. In order to mitigate this issue, researchers propose a strategy in their article [22] that uses a system-level data migration service provided by Android. In their proposed strategy, the system-level backup is taken from the target device which is an unrooted form to an intermediate device that is rooted. Data is then acquired from the rooted device to the workstation which would not have been possible directly from the unrooted target device.

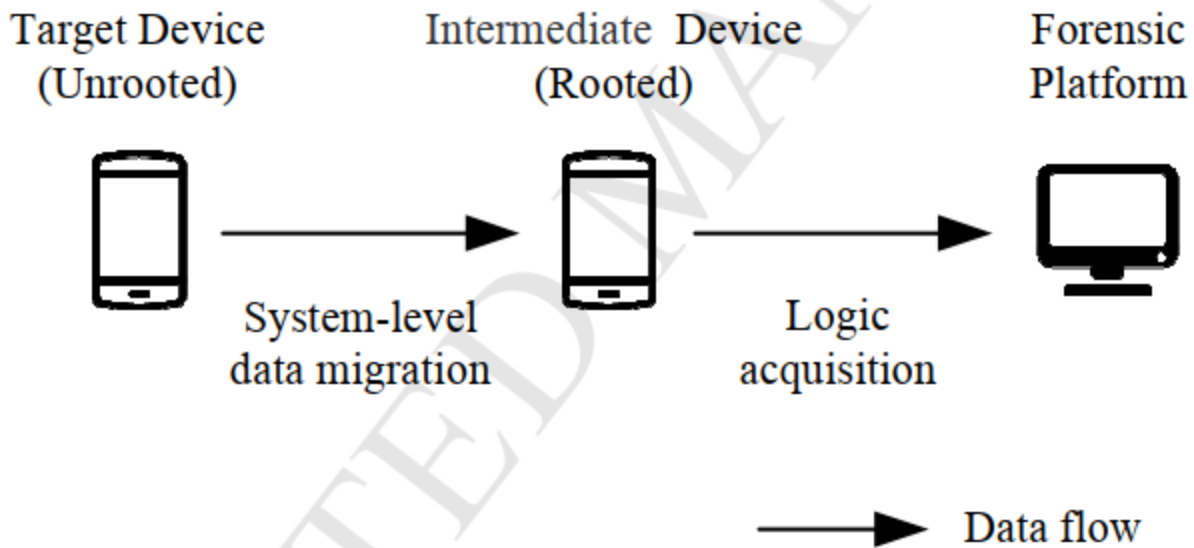


Figure 2-2 Data Acquisition through an Intermediate Device [22]

Mobile devices now support file-based encryption where each file is encrypted with a unique key as opposed to the traditional disk-based encryption. This created a hurdle in the forensics process. In their article [23] the researchers have explored various methodologies of data acquisition which include the logical and physical acquisition. They have also overviewed the possible mechanisms of data decryption for forensic analysis purposes. Finally, they discussed the data integrity and legal issues surrounding the data decryption process. Cosimo Anglano et al. [24] developed a tool, called AnForA, which is used for automated forensic analysis of android applications. Similarly, Qian Luo et al. [25] also developed an automated forensics tool targeted towards android applications for children.

### 2.3 Challenges of Mobile Device Forensics

Numerous mobile device models, having different operating systems, which is just one of the problems forensic investigators face when finding artifacts from mobile devices. In their research, Sundar Krishnan et al. [26] discussed various factors that pose a challenge to the forensic process. These primarily included password-protected devices and encrypted data on mobile devices. Another issue that needs to be tackled in mobile device forensics is how to acquire data from the volatile memory, RAM. Bin Liu et al. [27] propose a memory acquisition method for android application forensics that provides you with a full memory mirror of your desired process.

Obfuscation of evidence is another challenge that is discussed by Xiaolu Zhang et al. [28] along with the de-obfuscation techniques and their impact on the investigations. Human

error also sometimes becomes a weak link in the forensic process. Nina Sunde et al. [29] argue how cognitive and human factors can cause evidence to become misleading. While acquiring data from a mobile device, it is important that the data on the mobile device is not altered. Lazaro A Herrera [30] discusses how to seize and acquire a mobile device for a forensic investigation while causing minimum loss to the data inside it.

## 2.4 Summary of Literature Review

Paper	Problem Statement	Findings	Limitations	Tools
<b>Forensics study of IMO call and chat app [1] – 2018</b>	Mobile application of IMO on both Android and iOS platforms was analyzed	Server links were found which could be accessed without authentication	Mobile device should be rooted/jailbroken for physical image	Helium Backup Utility, DB Browser, Cydia
<b>Encrypted Network Traffic Analysis of Secure Instant Messaging Application: A case Study of Signal Messenger App [4] – 2021</b>	The communication is end to end encrypted, therefore, forensic analysis becomes a challenge, therefore the researchers opted for a network forensics strategy	By studying the payloads, they were able to identify text, audio, video and image messages as well as calls	Lack of device forensic analysis	Wireshark, CISCO switch, Firewall
<b>Logical Acquisition Method Based on Data Migration for Android Mobile Devices [22] – 2018</b>	This paper proposes a system level data migration strategy to conduct mobile forensics on an unrooted mobile device	System level backup is taken from the target device to an intermediate device which is rooted and subsequent forensic analysis is conducted on that	Device manufacturer should provide system level data migration tools	System level data migration tools such as smart switch
<b>Forensic Analysis of TOR Browser on Windows 10 and Android 10 Operating Systems [15] – 2021</b>	The TOR project claims that privacy of the highest nature is offered to the users, however, the researcher disproved this claim	They were able to discover significant artifacts that revealed details about the user's browsing activities	Network forensics is neglected in this research	FTK Imager, Belkasoft, Frida

*Table 2-1 Summary of Literature Review*

## Chapter – 3

### 3 Research Methodology

The purpose of this chapter is two-fold. Firstly, we will be discussing the research methodology employed for mobile device forensics. Secondly, we will be going over the experimentation setup required to execute the forensics process.

#### 3.1 Mobile Devises Forensic Methodology

There are two main sources of guidelines when it comes to forensics methodology. These are NIST and ISO which are globally recognized bodies. Going over the respective forensics guidelines from both organizations, we find some common ground and the guidelines can be grouped into four stages, which are:

- Preservation
- Acquisition
- Examination & Analysis
- Reporting

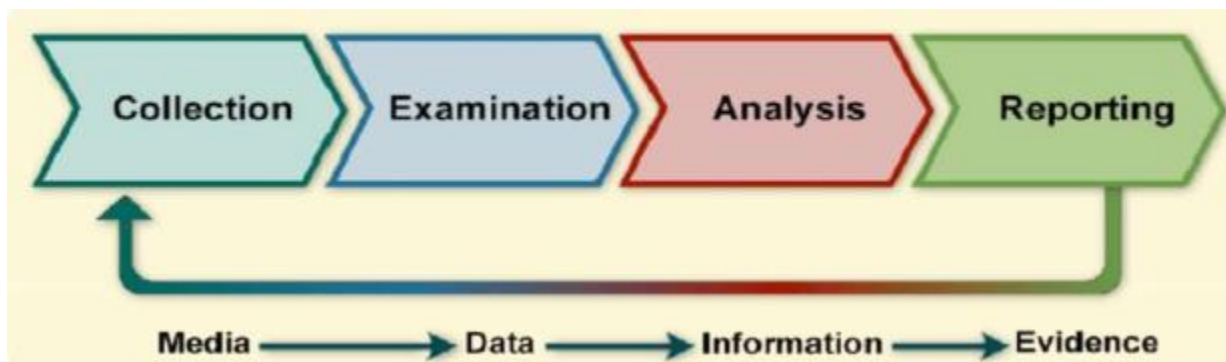


Figure 3-1 NIST's Forensic Model

We will now be discussing each stage in detail.

##### 3.1.1 Preservation

Preservation is the first step of digital evidence recovery. The purpose of preservation is to ensure that the contents of data are not altered and the custody of property is properly maintained. Preservation is the only thing that prevents evidence from being compromised and becoming ineligible in the court of law.

An important aspect of preservation is securing and evaluating the crime scene. Furthermore, the crime scene also needs to be documented which includes identifying and labeling non-electrical artifacts as well. Next, we come towards isolating the mobile



device to prevent network interference. Most mobile devices now come with the feature of remote master reset. To prevent that, network-level isolation should be considered immediately. After seizing the mobile device and isolating it, it should be packed and labeled securely. The environmental factors should be considered such as humidity and temperature.

### **3.1.2 Acquisition**

Acquisition refers to the process of extracting data from a mobile device for subsequent analysis. In order to acquire data efficiently, we first need to identify the model, make, and service provider of the mobile device. This can be established using the IMEI, FCC ID, and similar mechanisms. This information will allow us to choose the relevant tool for acquisition. The tool should be accurate, usable, comprehensive, and verifiable. With the help of the tool, we can acquire the data using either logical or physical images. A physical image takes more space on the hard disk but it provides more detailed information including deleted files.

### **3.1.3 Examination & Analysis**

Examination and analysis are closely linked processes but there is a fine line that differentiates the two. Examination refers to filtering out the relevant information whereas analysis provides insight on the basis of the results of the examination stage. An examination is more related to the technical domain as opposed to analysis which is a part of the forensics domain.

### **3.1.4 Reporting**

Reporting involves documenting each step performed with the help of screenshots, tool-generated content, and notes by the investigator. A good report always includes the inferences used to conclude. These inferences may be challenged in the court therefore they must be logically justified in the report. Finally, a report should not only include the software-generated content but also the analysis and findings of the investigator.

## **3.2 Experimentation Setup**

In this section, the following areas will be discussed:

- The software tools required for acquiring the data, examining it, and conducting our analysis

- Preparation of the workstation and mobile device for conducting forensic analysis
- Procedure for rooting the android mobile device
- How to acquire the data from the mobile device and transfer it to the workstation

Following is the strategy that will be followed while attempting this research project:

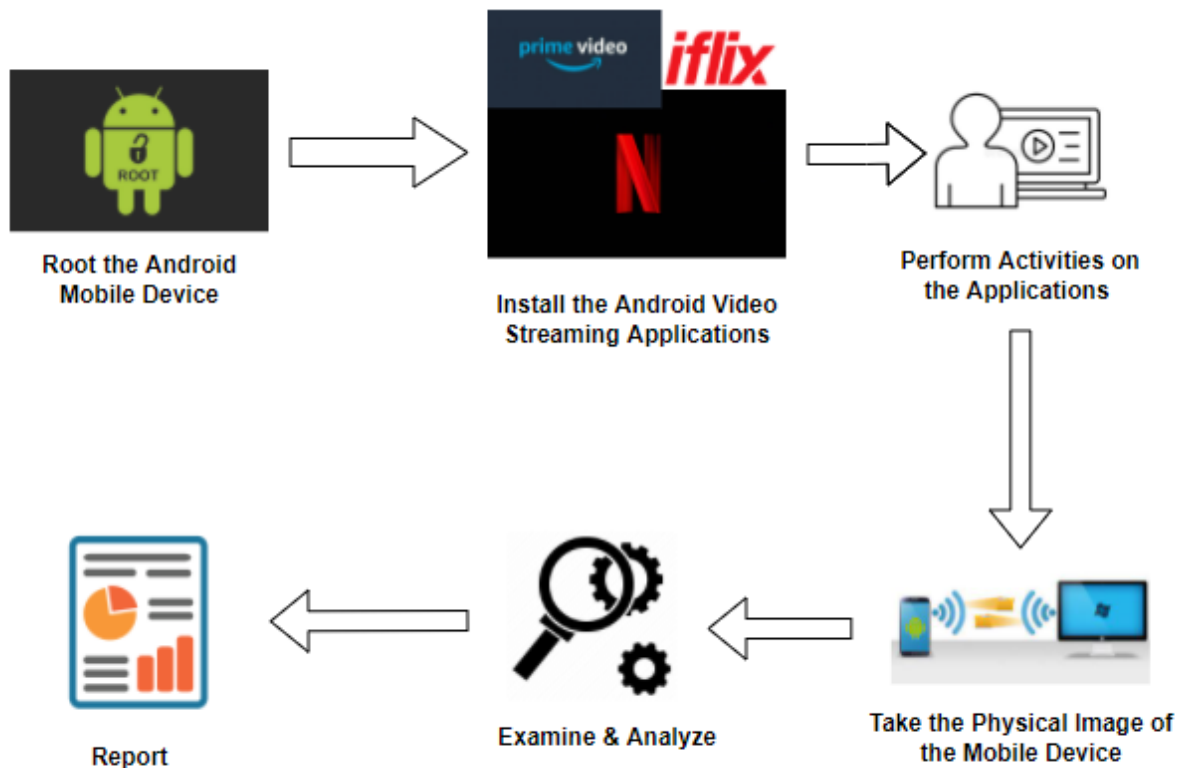


Figure 3-2 Proposed Methodology

### 3.2.1 Tools & Technologies

Here is the list of software tools that were installed on our workstation and mobile to conduct the experimentation process:

#### Workstation:

- Android Debug Bridge (ADB)
- Netcat (as part of the NMAP 7.92 bundle)
- Autopsy (4.19.1)
- DCode (5.5.21194.40)
- Odin (3.14.1)

### Mobile Device:

- Busybox
- Root Browser
- Root Checker
- Magisk
- Magisk Manager

### **3.2.2 Environment Readiness**

Environment readiness is a fundamental procedure conducted prior to the data acquisition phase. It is the first step of the experimentation process and is a prerequisite for later stages. During environment readiness, we prepare the workstation, mobile device as well as the communication channel between the two. We will first be discussing how to prepare the workstation followed by the steps required to install the specific tools on the mobile device.

### Workstation Readiness:

The specification of the workstation is as follows:

- HP Pavilion 15
- Intel Core i7, 2.4 GHz Dual-Core Processor
- 8 GB RAM
- Windows 10 Pro
- 64 Bit Operating System

The following software tools were installed:

- ADB
  - Android Debug Bridge (ADB) is a command-line tool used to communicate with android devices via the workstation. It is used to install, uninstall and debug applications on the android device through the workstation. Furthermore, in our scenario, ADB aids is the acquisition of the data from the android mobile device.
  - In order to install ADB on the workstation, we first need to download the SDK-Platform tools for Windows through the following link:  
<https://developer.android.com/studio/releases/platform-tools>

- After it has finished downloading, extract the compressed file
- In order to use the ADB command on the command prompt from any path, add the path of ADB to the environment variables
- In order to add the path of ADB to environment variables, go to “Advanced System Settings” and click on “Environment Variables”
- On the “Environment Variables” window, edit the “Path” variable and add the path of ADB to it
- To verify successful addition of ADB's path to environment variables, open command prompt from the default home path and enter “ADB”
- All possible commands of the android debug bridge will show up
- Netcat
  - Netcat is a computer networking tool used to read from and write to network connections using the TCP and UDP protocols.
  - In our scenario, we use Netcat to transfer data from the mobile device to the workstation using a TCP based connection
  - In order to install Netcat, we need to download and install the NMAP bundle from the following link: [www.nmap.org](http://www.nmap.org)
  - Netcat comes as a part of the NMAP bundle
- Autopsy
  - Autopsy is a graphical forensic tool used to examine and analyze data
  - Visit the following link: <https://www.autopsy.com/download/>
  - Download Autopsy for Windows
  - Follow the on-screen instructions for installation
- DCode

- DCode is a forensic utility used to convert raw hex data into human-readable timestamps
- Download DCode from the following link: [www.digita-detectives.net/dcode](http://www.digita-detectives.net/dcode)
- Odin
  - Odin is a utility used to communicate with Samsung mobile devices through a workstation
  - It is used to flash recovery firmware image to a Samsung android device
  - It can be downloaded from the following link: [www.odindownload.com](http://www.odindownload.com)

### Mobile Device Readiness:

The specifications of the mobile device are as follows:

- Samsung Galaxy Note 5
- Android 7.0
- 32 GB Internal Storage
- 4 GB RAM

The following software tools were installed:

- Root Checker
  - It can be installed from the google play store and it is used to verify the successful completion of the rooting procedure of the android mobile device.
- Root Browser
  - It is used to view and modify system files that can only be accessed from a rooted device. This can be downloaded from the google play store.
- Busy Box
  - This is also installed from the google play store and it is used to install the “dd” utility on the android mobile device. This utility is used to transfer data from the mobile device to the workstation, especially during physical image acquisition.
- Magisk Manager

It is used to manage a rooted device and control the root privileges for a different types of applications. This application, however, is not available on the app store and we need to manually download the apk from the following link: [www.odindownload.com/magisk-manager](http://www.odindownload.com/magisk-manager). Once the apk has been downloaded, go ahead and install the apk on the mobile device. You may receive a warning message but you should allow installation of apk downloaded from an unknown source.

- **Magisk**

This application is a requirement of the rooting procedure on the android device. It is not available on the google play store, however, you can download its apk from the following link: [www.magiskmanager.com](http://www.magiskmanager.com). Once the apk has been downloaded, you can install it on your device.

### **3.2.3 Rooting Process**

The following steps were followed to root the android mobile device:

- Download the respective TWRP recover file in the .tar format for Samsung Galaxy Note 5 from the following link: [www.twrp.me/devices](http://www.twrp.me/devices)
- Download and install Magisk and Magisk Manager
- Enable developer mode on your mobile device by pressing the build number in settings 7 times
- Enable OEM and USB debugging options on the android mobile device
- Reboot the phone in recovery mode, by pressing the Volume Down, Power, and Home button simultaneously. Release all the keys when the recovery mode appears
- Connect the device with the workstation, open the Odin software and disable the auto-reboot option
- Next, click on AP and select the TWRP file

- Press the Volume Up key on your mobile device and press Start on the Odin Software
- Press Volume Down, Home, and Power Key to get out of recovery mode. Once a blank screen appears press the Volume Up key instead of the Volume Down key.
- Release all keys when TWRP recovery appears.
- Select Wipe and Format data. Reboot to recovery after that.
- Copy Magisk apk on to the phone by connecting the phone with the workstation and install the application using the TWRP recovery interface. Reboot to system after that.
- Setup your phone and verify that the device has been rooted successfully using the root checker application. You will have to grant it root privileges in order for the application to work properly.

### 3.2.4 Data Acquisition Method

For acquiring the data from the mobile device, physical acquisition methodology was preferred as it enables us to acquire a bit-by-bit image of a mobile device. The “dd” utility and Netcat were used to perform the physical acquisition of the mobile device data onto the workstation. The following steps were carried out to perform the physical acquisition:

- Acquire the list of android mobile devices attached to the workstation using the ADB command

```
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\PC>adb devices
List of devices attached
8575484242565931      device
```

- Establish the connection between the workstation and the mobile device. Next, list of files in the /dev/block directory

```
C:\Users\PC>adb shell
noblelte:/ # cd /dev/block
noblelte:/dev/block # ls
loop0      persistent ram15      sda        sda17      sdb
loop1      platform   ram2       sda1       sda2       sdc
loop2      ram0       ram3       sda10      sda3       sdd
loop3      ram1       ram4       sda11      sda4       sdd1
loop4      ram10      ram5       sda12      sda5       steady
loop5      ram11      ram6       sda13      sda6       vnswap0
loop6      ram12      ram7       sda14      sda7       vold
loop7      ram13      ram8       sda15      sda8
param      ram14      ram9       sda16      sda9
noblelte:/dev/block # cd ~
```

- Open a new command prompt and define the ports of the TCP connection for the data session. Make the configurations such that both the listening port and forwarding port is set to 8888

```
Microsoft Windows [Version 10.0.19043.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Users\PC>adb forward tcp:8888 tcp:8888
8888
```

- Setup a listening connection on the phone at port 8888, Busybox will wait for a TCP connection at port 8888 and once the connection has been established with the workstation it will forward the respective file

```
noblelte:/ # dd if=/dev/block/sda | busybox nc -l -p 8888
```

- Initiate a connection from the workstation at port 8888 and provide the path at which you want to store the received file

```
C:\Users\PC>ncat.exe 127.0.0.1 8888 >D:\image1.dd
```

- Wait for data transfer confirmation

```
noblelte:/ # dd if=/dev/block/sda | busybox nc -l -p 8888
62464000+0 records in
62464000+0 records out
31981568000 bytes (29.8GB) copied, 1539.600975 seconds, 19.8MB/s
noblelte:/ #
```



## Chapter – 4

### 4 Applications Examination & Analysis

This chapter will include the examination and analysis of the three android video streaming applications, which are the following:

- Netflix
- Amazon Prime Video
- iFlix

The examination and analysis have been performed on the physical images acquired from the mobile device after performing every activity. The following are the main groups of activities:

- Application Installation
- User Login
- Viewing a Video
- Searching for a Video
- Creating a List of Videos
- Downloading a Video
- Creating User Profiles
- Rating a Video
- Deleting a User Profile
- Uninstalling the Application

<b>Code of Groups of Activities</b>	<b>Description of the Group</b>
AI	Downloading the application from Google Play Store or the internet and the subsequent Installation
UL	Logging in to the application
VV	Viewing an episode of a TV Show or watching a Movie
SV	Entering a search query in the search bar
CL	Creating a list of videos for easier access
DV	Downloading a video for later use
CP	Creating a separate profile for another user
RV	Providing feedback by upvoting or downvoting a video
DP	Deleting a user profile that was earlier created
UA	Uninstalling the application from the mobile device

*Table 4-1 Description of the Codes of Groups of Activities*

## 4.1 Netflix

Netflix is the oldest and most popular online video streaming platform. It can be accessed both on the web and through mobile device applications. The company was formed in 1997 when it operated as a DVD rental business. It began its online streaming services in 2007 and ever since then it has seen tremendous success and growth. As of the year 2020, Netflix has over 200 million paid subscribers.

Netflix is also available on android mobile devices where it has over a billion downloads. In order to subscribe to the streaming platform, you need to purchase the subscription via a debit or credit card. There are multiple subscription plans available depending upon your requirements. The cheapest plan only allows you to view the content on your smartphones and tablets in 480p resolution, whereas the most expensive allows you to watch on any device with resolutions up to 4k.

The analysis of this application is based on its version 8.6.0, whose apk is available online at the following link: [www.apkmirror.com](http://www.apkmirror.com)

### 4.1.1 Remnants of AI

Netflix cannot be downloaded on a rooted android mobile device via google play store, however, we can download the apk from the link provided earlier. Upon installation, the application still refuses to work even if you hide the fact that the device has been rooted using the Magisk Manager application. The following steps are required in order to be able to properly interact with the Netflix android application:

- Open the root browser (after following the installation instructions provided earlier)
- Navigate to /system/lib folder
- Search for the file named “liboemcrypto.so”
- Rename the file to “liboemcrypto.so.bak”
- This will make the file inactive
- Restart the phone and now Netflix will function properly

Following are the remnants discovered after installing the downloaded apk, which are grouped by the path at which they were found:

#### **/app**

1. /com.netflix.mediaclient-1

Name	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)	Known	Location
com.netflix.mediaclient-1	2021-10-27 15:06:07 PKT	2021-10-27 15:06:30 PKT	2021-10-27 15:05:58 PKT	2021-10-27 15:05:58 PKT	4096	Allocated	unknown	/img_image1.dd/vol_vol20/app/com.netflix.mediaclient-1

This is the native location for the libraries of all the downloaded applications. The screenshot above shows that the application was downloaded at 3:05:58 PM PKT on 27<sup>th</sup> Oct 2021.

## /data

### 1. /com.netflix.mediaclient

Name	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)	Known	Location
com.netflix.mediaclient	2021-10-27 15:06:56 PKT	2021-10-27 15:06:56 PKT	2021-10-27 15:06:31 PKT	2021-10-27 15:06:31 PKT	4096	Allocated	unknown	/img_image1.dd/vol_vol20/data/com.netflix.mediaclient

This folder is created soon after the native folder is created in the /app directory. The time of the creation of this folder is 3:06:31 PM PKT, the slight delay shows that the folder was first created in the /app directory and then in the /data directory.

### 2. /com.android.vending/databases/frosting.db

pk	last_updated	frosting_id	apk_path	data
com.netflix.mediaclient	1635329190983	0	/data/app/com.netflix.mediaclient-1/base.apk	BLOB Dat...

Frosting database contains a table called “frosting” which includes the package name, last updated time, frosting ID, and the path to the apk.

The last updated time is given in raw format, we can convert to human-readable timestamp format using the DCode application. Following screenshot shows the conversion:

⌚ Unix Milliseconds (Java Time)	2021-10-27 15:06:30.983 +05:00
---------------------------------	--------------------------------

### 3. /com.google.android.gms/databases

#### a) gass.db

gass.db					2021-10-27 18:29:29 PKT	2021-10-27 18:59:29 PKT	2021-10-24 12:43:16 PKT	2021-10-24 12:43:16 PKT	36864										
<div style="display: flex; justify-content: space-between; align-items: center;"> <span>&lt;</span> <span>Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences</span> </div> <div style="display: flex; justify-content: space-between; align-items: center; margin-top: 5px;"> <span>Table <b>app_info</b> 51 entries Page 1 of 1</span> <span>Export to CSV</span> </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>_id</th> <th>pb</th> <th>package_name</th> <th>version_code</th> <th>digest_sha256</th> </tr> </thead> <tbody> <tr> <td>43</td> <td>BLOB Data not shown</td> <td>com.netflix.mediaclient</td> <td>40047</td> <td>24c40a640434362368f2a9791f33b90eefbc63fb009617d74e53de73f7832997</td> </tr> </tbody> </table>										_id	pb	package_name	version_code	digest_sha256	43	BLOB Data not shown	com.netflix.mediaclient	40047	24c40a640434362368f2a9791f33b90eefbc63fb009617d74e53de73f7832997
_id	pb	package_name	version_code	digest_sha256															
43	BLOB Data not shown	com.netflix.mediaclient	40047	24c40a640434362368f2a9791f33b90eefbc63fb009617d74e53de73f7832997															

In the Gass database, there is a table called “app\_info” which contains the information about the installed android applications. The Netflix package along with its package name, version, and package hash was found. The hashing algorithm used to determine the package hash is SHA-256.

b) google\_app\_measurement.db

app_id	app_instance_id	app_version	app_store	gmp_ver.
com.netflix.mediaclient	9974f64f05736593f00ef9d08627266d	8.6.0 build 2 40047	com.google.android.packageinstaller	19000

This database contains a table called “apps” from where we can determine the application ID, its instance ID as well its version

**/user\_de**

1. /0/ com.netflix.mediaclient

Name	Modified Time	Change Time	Access Time	Created Time
com.netflix.mediaclient	2021-10-27 15:06:35 PKT	2021-10-27 15:06:35 PKT	2021-10-27 15:06:31 PKT	2021-10-27 15:06:31 PKT

At this location, we found a folder with Netflix’s package name and its creation time corresponds with the installation of the application.

### 4.1.2 Remnants of UL

You can sign up and subscribe using the android application of Netflix and if you are already a subscriber, you can just enter your credentials and login to the application.

Following are the remnants discovered while logging in to Netflix application:

**/data**

1. /com.samsung.android.providers.context/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
209	com.netflix.mediaclient	com.netflix.mediaclient.acquisition2.screens.signupContainer.SignupNativeActivity	2021-10-27 10:09:17.223	2021-10-27 10:09:33.625
210	com.netflix.mediaclient	com.netflix.mediaclient.ui.login.LoginActivity	2021-10-27 10:09:33.654	2021-10-27 10:09:36.989

This database contains a table called “use\_app” which contains the artifacts from the signing up and logging inactivity along with the timestamps. These timestamps are in GMT (Greenwich Mean Time) zone.

### 4.1.3 Remnants of VV

#### /data

1. /com.netflix.mediaclient/databases/appHistory

playableId	xid	eventTime	eventType	network	duration	offline	id
80166369	7024062107100733554	1635417156292	1	1	8014000	0	1

This database has a table called “playEvent” which stores information about all the videos played on the application whether online or offline. Here we can see that a video was played along with its timestamp and duration, furthermore, we can also identify that the video was played online, as its offline flag is set to 0.

The timestamp is in raw format, which when converted to human-readable format, gave the following output:

Unix Milliseconds (Java Time) 2021-10-28 15:32:36.292 +05:00

2. com.samsung.android.providers.contexty/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
317	com.netflix.mediaclient	com.netflix.mediaclient.ui.player.PlayerActivity	2021-10-28 10:28:21.077	2021-10-28 10:32:36.205

In the table “use\_app”, we can view the artifact for a video viewing activity along with its starting and finishing time given according to GMT zone.

### 4.1.4 Remnants of SV

/data

1. com.samsung.android.providers.contexty/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
314	com.netflix.mediaclient	com.netflix.mediaclient.ui.search.PortraitSearchActivity	2021-10-28 10:23:16.604	2021-10-28 10:23:58.799

The table named “use\_app” contains information about all the activities performed in an application. Here we can see the artifact for a search activity conducted along with its timestamp.

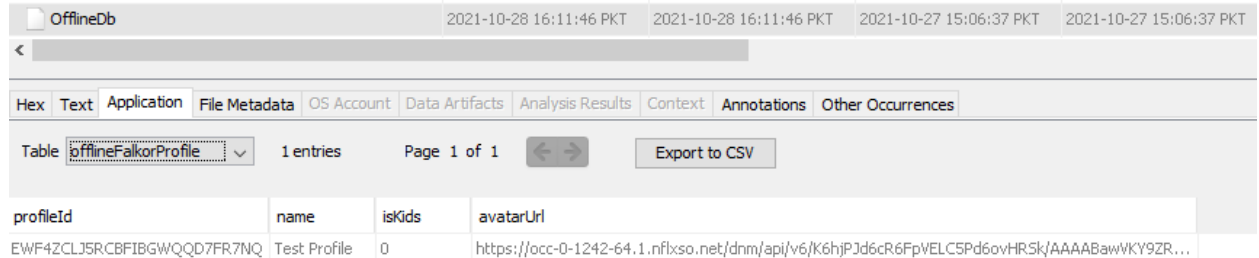
### 4.1.5 Remnants of CL

No remnants were found for this activity.

## 4.1.6 Remnants of DV

/data

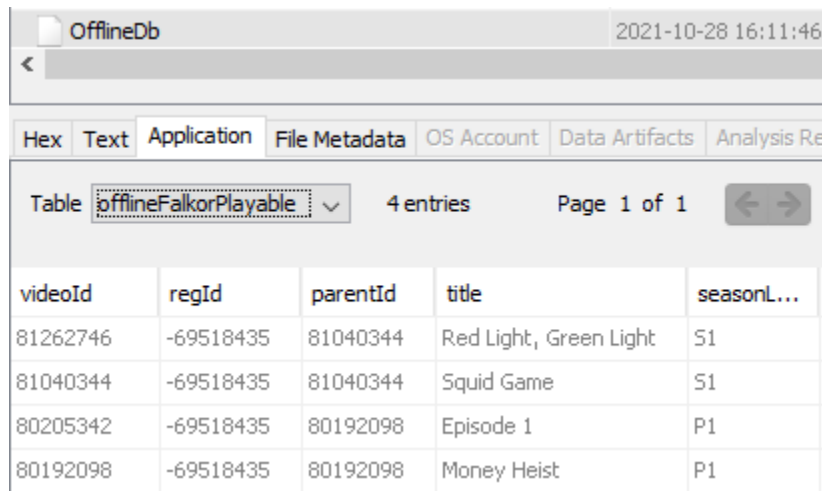
1. /com.netflix.mediaclient/databases



The screenshot shows a database viewer interface for 'OfflineDb'. The table 'offlineFalkorProfile' is selected, showing 1 entry on page 1 of 1. The table has columns: profileId, name, isKids, and avatarUrl. The data row shows profileId: EWF4ZCLJ5R3CBFIBGWQD7FR7N9, name: Test Profile, isKids: 0, and avatarUrl: https://occ-0-1242-64.1.nflxso.net/dnm/api/v6/K6hjPJd6cR6FpVELC5Pd6ovHR5k/AAAAABawWKY9ZR...

profileId	name	isKids	avatarUrl
EWF4ZCLJ5R3CBFIBGWQD7FR7N9	Test Profile	0	https://occ-0-1242-64.1.nflxso.net/dnm/api/v6/K6hjPJd6cR6FpVELC5Pd6ovHR5k/AAAAABawWKY9ZR...

In the table “offlineFalkorProfile”, we can view information about the profile through which the video was downloaded. Here, we can view in plaintext the name of the profile which is “Test Profile”, but in real-life scenarios, it will most probably be the name of the user.



The screenshot shows a database viewer interface for 'OfflineDb'. The table 'offlineFalkorPlayable' is selected, showing 4 entries on page 1 of 1. The table has columns: videoId, regId, parentId, title, and seasonL... The data rows show videoId, regId, parentId, title, and seasonL... for four entries.

videoId	regId	parentId	title	seasonL...
81262746	-69518435	81040344	Red Light, Green Light	S1
81040344	-69518435	81040344	Squid Game	S1
80205342	-69518435	80192098	Episode 1	P1
80192098	-69518435	80192098	Money Heist	P1


In the table “offlineFalkorPlayable”, we can view information about the videos downloaded. Here we can see that we have downloaded two episodes of two different TV Shows. The first episode is titled “Red Light, Green Light” from the TV Show “Squid Game”. The second is titled “Episode 1 from the TV Show “Money Heist”.

## 2. /com.netflix.mediaclient/files/img/of/videos

80192098.img	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 15:17:42 PKT	2021-10-28 15:17:42 PKT	43044	Allocated	unknown
80205342.img	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	51828	Allocated	unknown
81040344.img	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 15:17:39 PKT	2021-10-28 15:17:39 PKT	31406	Allocated	unknown
81262746.img	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	27910	Allocated	unknown

Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences


0° 88% Reset



80192098.img	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 15:17:42 PKT	2021-10-28 15:17:42 PKT	43044	Allocated	unknown	/i
80205342.img	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	2021-10-28 16:04:15 PKT	51828	Allocated	unknown	/i
81040344.img	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 15:17:39 PKT	2021-10-28 15:17:39 PKT	31406	Allocated	unknown	/i
81262746.img	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	2021-10-28 16:03:14 PKT	27910	Allocated	unknown	/i

Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences

0° 88% Reset



In this location, we can find the title images of the TV Shows, whose episodes we have downloaded.



### 4.1.7 Remnants of CP

No remnants were found for this activity.

### 4.1.8 Remnants of RV

No remnants were found for this activity.

### 4.1.9 Remnants of DP


No remnants were found for this activity.

### 4.1.10 Remnants of UA

The Netflix application is finally uninstalled from the android mobile device. These artifacts related to the Netflix application still remained after it had been uninstalled from the mobile device:

#### **/app**


1. /com.netflix.mediaclient-1

Name	Modified Time	Change Time	Access Time	Created Time	Size
 com.netflix.mediaclient-1	2021-10-29 15:27:29 PKT	2021-10-29 15:27:29 PKT	2021-10-27 15:05:58 PKT	2021-10-27 15:05:58 PKT	0

Here we can see the deleted folder of “com.netflix.mediaclient-1” which shows that the Netflix application was once installed but now it is deleted.

#### **/data**

1. /com.netflix.mediaclient

Name	Modified Time	Change Time	Access Time	Created Time	Size
 com.netflix.mediaclient	2021-10-29 15:27:27 PKT	2021-10-29 15:27:27 PKT	2021-10-27 15:06:31 PKT	2021-10-27 15:06:31 PKT	0

Here again, we can see the deleted folder of “com.netflix.mediaclient” which shows that the Netflix application was once installed but now it is deleted.

## 2. /com.google.android.gms/databases/gass.db

_id	package_name	version_code	digest_sha256
43	com.netflix.mediaclient	40047	24c40a640434362368f2a9791f33b90eefbc63fb009617d74e53de73f7832997

This database has a table called “app\_info” which contains information about all the applications installed on the android mobile device even after they have been uninstalled.

## 3. /com.samsung.android.providers.context/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
314	com.netflix.mediaclient	com.netflix.mediaclient.ui.search.PortraitSearchActivity	2021-10-28 10:23:16.604	2021-10-28 10:23:58.799
315	com.netflix.mediaclient	com.netflix.mediaclient.ui.search.PortraitSearchActivity	2021-10-28 10:26:21.024	2021-10-28 10:27:18.550
316	com.netflix.mediaclient	com.netflix.mediaclient.ui.search.PortraitSearchActivity	2021-10-28 10:28:16.206	2021-10-28 10:28:20.832
317	com.netflix.mediaclient	com.netflix.mediaclient.ui.player.PlayerActivity	2021-10-28 10:28:21.077	2021-10-28 10:32:36.205
318	com.netflix.mediaclient	com.netflix.mediaclient.ui.search.PortraitSearchActivity	2021-10-28 10:32:36.399	2021-10-28 10:32:39.765

All the activities performed using the application are still visible in the “use\_app” table even after uninstalling the application.

### 4.1.11 Summary of Remnants

Code of Groups of Activities	Main Directory	Folder	Filename
1) AI	1.1)/app	1.1.1)/com.netflix.mediaclient -1	1.1.1-a) Main Folder
	1.2)/data	1.2.1)/com.netflix.mediaclient	1.2.1-a) Main Folder
		1.2.2)	1.2.2-a) frosting.db
	1.2.3-a) gass.db	/com.android.vending/databases/	1.2.3-a) gass.db
1.2.3)		/com.google.android.gms/databases	1.2.3-b) google_app_measurement .db
1.3)/user_de	1.3.1) /0/com.netflix.mediaclient	1.3.1-a) Main Folder	

2) UL	2.1)/data	2.1.1)/com.samsung.android.providers.context/databases	2.1.1-a) ContextLog.db
3) VV	3.1)/data	3.1.1) /com.netflix.mediaclient/databases 3.1.2) /com.samsung.android.providers.context/databases	3.1.1-a) appHistory 3.1.2-a) ContextLog.db
4) SV	4.1)/data	4.1.1) /com.samsung.android.providers.context/databases	4.1.1-a) ContextLog.db
5) CL	-	-	-
6) DV	6.1)/data	6.1.1) /com.netflix.mediaclient/databases 6.1.2) /com.netflix.mediaclient/files /img/of/videos	6.1.1-a) offlineDb 6.1.2-a) Image Files
7) CP	-	-	-
8) RV	-	-	-
9) DP	-	-	-
10) UA	4.1)/app	4.1.1) /com.netflix.mediaclient -1	4.1.1-a) Main Folder
	4.2)/data	4.2.1) /com.netflix.mediaclient 4.2.2) /com.google.android.gms/databases 4.2.3)/com.samsung.android.providers.context/databases	4.2.1-a) Main Folder 4.2.2-a) gass.db 4.2.3-a) ContextLog.db

*Table 4-2 Summary of Netflix Remnants with Location Path*

## 4.2 Amazon Prime Video

Amazon Prime Video was launched in the year 2006 as an online video streaming service. Later its mobile applications were also released. Currently, it has over 175 million users and its android application has over 100 million downloads. Amazon Prime Video is available in over 200 countries worldwide.

Upon subscription, Amazon Prime Video offers you a one-week trial period during which you can cancel the subscription and you will not be charged for it. You can subscribe through the android application by using your credit card or through the google play store payments feature.


The analysis of Amazon Prime Video's android application is based on its version 3.0.308.15647, which is the latest version available on the google play store as of now.

## 4.2.1 Remnants of AI

The application can be downloaded and installed from the google play store even if the device is rooted. Following are the remnants discovered after downloading the application, which are grouped by the path at which they were found:

### /app


1. /com.amazon.avod.thirdpartyclient-1

Name	Modified Time	Change Time	Access Time	Created Time	Size
 com.amazon.avod.thirdpartyclient-1	2021-10-30 13:44:31 PKT	2021-10-30 13:44:43 PKT	2021-10-30 13:44:11 PKT	2021-10-30 13:44:11 PKT	4096

This is the native location for the libraries of all the downloaded applications. The screenshot above shows that the application was downloaded at 1:44:11 PM PKT on 30<sup>th</sup> Oct 2021.

### /data



1. /com.amazon.avod.thirdpartyclient

Name	Modified Time	Change Time	Access Time	Created Time	Size
 com.amazon.avod.thirdpartyclient	2021-10-31 12:17:22 PKT	2021-10-31 12:17:22 PKT	2021-10-30 13:44:43 PKT	2021-10-30 13:44:43 PKT	4096

This folder is created soon after the native folder is created in the /app directory. The time of creation of this folder is 1:44:43 PM PKT, the slight delay shows that the folder was first created in the /app directory and then in the /data directory.

2. /com.android.vending/databases

- a) frosting.db

frosting.db		2021-10-31 12:00:04 PKT	2021-10-31 12:00:04 PKT	2021-10-24 12:49:52 PKT	2021-10-24 12:49:52 PKT
					
Hex Text Application File Metadata OS Account Data Artifacts Analysis Results Context Annotations Other Occurrences					
Table frosting		332 entries	Page 4 of 4	 <a href="#">Export to CSV</a>	
pk	frosting_id	last_u...	apk_path	data	
com.amazon.avod.thirdpartyclient	1	16355834...	/data/app...	BLOB Dat...	

Frosting database contains a table called “frosting” which includes the package name, last updated time, frosting ID and the path to the apk.

b) library.db

The screenshot shows a database viewer interface for 'library.db'. The table 'ownership' is selected, showing 734 entries on page 8 of 8. An 'Export to CSV' button is visible. The table has the following columns and one data row:

account	library_id	backend	doc_id	doc_type	offer_type	documen...	subs_vali...	app_cert...	app_refu...	app_refu...	subs_aut...
adilahmad95@gmail.com	3	3	com.amazon.avod.thirdpartyclient	1	1	58350367...		oYNSTDyP...	0	0	

The table called “ownership” contains information about all the packages downloaded from the google play store along with the account used to download them.

c) localappstore.db

The screenshot shows a database viewer interface for 'localappstore.db'. The table 'appstate' is selected, showing 42 entries on page 1 of 1. An 'Export to CSV' button is visible. The table has the following columns and one data row:

package_name	auto_up...	desired_...	downloa...	delivery_data	delivery_data_...	installer_...	first_download...	referrer	account
com.amazon.avod.thirdpartyclient	1	-1		BLOB Data not shown	1635583452083	0	1635583455691		adilahmad95@gmail.com

The table called “appstore” also shows information about the packages downloaded from the google play store.

3. /com.google.android.gms/databases/gass.db


The screenshot shows a database viewer interface for 'gass.db'. The table 'app\_info' is selected, showing 60 entries on page 1 of 1. An 'Export to CSV' button is visible. The table has the following columns and one data row:

_id	pb	package_name	version_code	digest_sha256
59	BLOB Data not shown	com.amazon.avod.thirdpartyclient	308015647	7f9c8d08306f9aa8ed8096345e78d0c3f757e67d1dd7023b7944bc72e2666c03

In the Gass database, there is a table called “app\_info” which contains the information about the installed android applications. The Amazon Prime Video package along with its package name, version, and package hash was found. The hashing algorithm used to determine the package hash is SHA-256.

## /user\_de

1. /0/com.amazon.avod.thirdpartyclient

Name	Modified Time	Change Time	Access Time	Created Time	Size
 com.amazon.avod.thirdpartyclient	2021-10-30 13:44:48 PKT	2021-10-30 13:44:48 PKT	2021-10-30 13:44:43 PKT	2021-10-30 13:44:43 PKT	4096

At this location we found a folder with Amazon Prime Video’s package name and its creation time corresponds with the installation of the application.

## 4.2.2 Remnants of UL

You can sign up and subscribe using the android application of Amazon Prime Video and if you are already a subscriber, you can just enter your credentials and login to the application.

Following are the remnants discovered while logging in to Amazon Prime Video application:

## /data

1. /com.samsung.android.providers.context/databases/ContextLog.db

ContextLog.db		2021-10-31 13:52:03 PKT	2021-10-31 13:52:03 PKT	2021-10-24 12:42:52 PKT	2021-10-24 12:42:52 PKT	131072	Allocated	unknown												
<div style="display: flex; justify-content: space-between; border-bottom: 1px solid #ccc; padding-bottom: 5px;"> <span>Hex</span> <span>Text</span> <span>Application</span> <span>File Metadata</span> <span>OS Account</span> <span>Data Artifacts</span> <span>Analysis Results</span> <span>Context</span> <span>Annotations</span> <span>Other Occurrences</span> </div> <div style="display: flex; justify-content: space-between; border-bottom: 1px solid #ccc; padding-bottom: 5px;"> <span>Table use_app</span> <span>544 entries</span> <span>Page 4 of 6</span> <span>◀ ▶</span> <span>Export to CSV</span> </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>_id</th> <th>app_id</th> <th>app_sub_id</th> <th>start_time</th> <th>stop_time</th> <th>starttime</th> </tr> </thead> <tbody> <tr> <td>394</td> <td>com.amazon.avod.thirdpartyclient</td> <td>com.amazon.avod.client.activity.PrimeSignUpActivity</td> <td>2021-10-30 08:47:12.636</td> <td>2021-10-30 08:47:36.704</td> <td>16355836...</td> </tr> </tbody> </table>									_id	app_id	app_sub_id	start_time	stop_time	starttime	394	com.amazon.avod.thirdpartyclient	com.amazon.avod.client.activity.PrimeSignUpActivity	2021-10-30 08:47:12.636	2021-10-30 08:47:36.704	16355836...
_id	app_id	app_sub_id	start_time	stop_time	starttime															
394	com.amazon.avod.thirdpartyclient	com.amazon.avod.client.activity.PrimeSignUpActivity	2021-10-30 08:47:12.636	2021-10-30 08:47:36.704	16355836...															

In the table called “use\_app” you can view the artifacts from the signing up activity

ContextLog.db				
2021-10-31 13:52:03 PKT	2021-10-31 13:52:03 PKT	2021-10-24 12:42:52 PKT	2021-10-24 12:42:52 PKT	131072
<div style="display: flex; justify-content: space-between;"> <span>Hex</span> <span>Text</span> <span>Application</span> <span>File Metadata</span> <span>OS Account</span> <span>Data Artifacts</span> <span>Analysis Results</span> <span>Context</span> <span>Annotations</span> <span>Other Occurrences</span> </div>				
Table use_app 544 entries Page 5 of 6 <span>Export to CSV</span>				
_id	app_id	app_sub_id	start_time	stop_time
402	com.amazon.avod.thirdpartyclient	com.android.billingclient.api.ProxyBillingActivity	2021-10-30 08:49:01.811	2021-10-30 08:49:01.848

Here we can see the artifacts from the billing activity along with the timestamps.

### 4.2.3 Remnants of VV

/data

1. /com.samsung.android.providers.context/databases/ContextLog.db

ContextLog.db				
2021-10-31 13:52:03 PKT	2021-10-31 13:52:03 PKT	2021-10-24 12:42:52 PKT	2021-10-24 12:42:52 PKT	1311
<div style="display: flex; justify-content: space-between;"> <span>Hex</span> <span>Text</span> <span>Application</span> <span>File Metadata</span> <span>OS Account</span> <span>Data Artifacts</span> <span>Analysis Results</span> <span>Context</span> <span>Annotations</span> <span>Other Occurrences</span> </div>				
Table use_app 544 entries Page 5 of 6 <span>Export to CSV</span>				
_id	app_id	app_sub_id	start_time	stop_time
462	com.amaz...	com.amazon.avod.thirdpartyclient.ThirdPartyPlaybackActivity	2021-10-31 07:17:14.914	2021-10-31 07:25:54.836

In the table called “use\_app”, we can see the artifact for an activity where the user watched a video. The starting and finishing time of the activity are also mentioned.

### 4.2.4 Remnants of SV

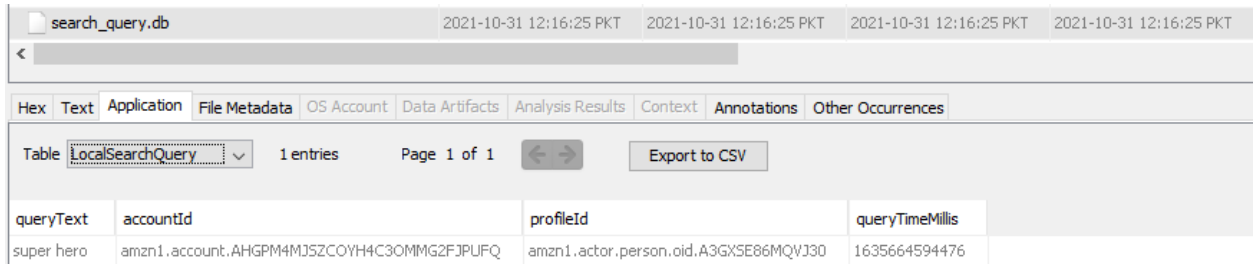
/data

1. /com.samsung.android.providers.context/databases/ContextLog.db

ContextLog.db				
2021-10-31 13:52:03 PKT	2021-10-31 13:52:03 PKT	2021-10-24 12:42:52 PKT	2021-10-24 12:42:52 PKT	131072
<div style="display: flex; justify-content: space-between;"> <span>Hex</span> <span>Text</span> <span>Application</span> <span>File Metadata</span> <span>OS Account</span> <span>Data Artifacts</span> <span>Analysis Results</span> <span>Context</span> <span>Annotations</span> <span>Other Occurrences</span> </div>				
Table use_app 544 entries Page 6 of 6 <span>Export to CSV</span>				
_id	app_id	app_sub_id	start_time	stop_time
506	com.amazon.avod.thirdpartyclient	com.amazon.avod.client.activity.SearchQueryActivity	2021-10-31 07:31:31.918	2021-10-31 07:31:45.176
508	com.amazon.avod.thirdpartyclient	com.amazon.avod.client.activity.SearchQueryActivity	2021-10-31 07:31:51.212	2021-10-31 07:31:52.322

In the same table, we can view the artifacts for search activity.

## 2. /com.amazon.avod.thirdpartyclient/databases/search\_query.db



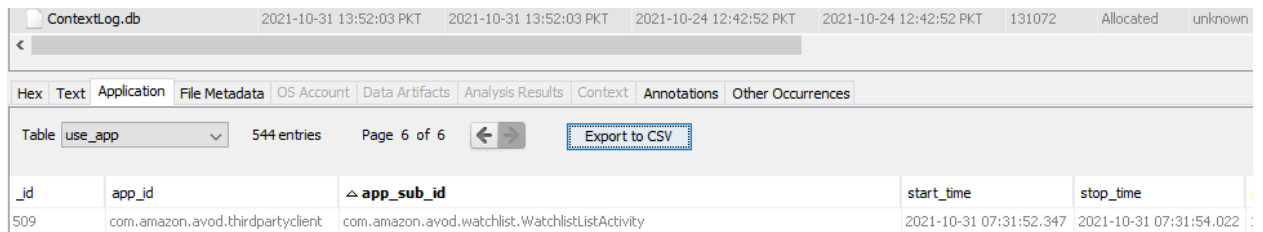
queryText	accountId	profileId	queryTimeMillis
super hero	amzn1.account.AHGPM4MJ5ZCOYH4C3OMMG2FJPUFQ	amzn1.actor.person.oid.A3GX5E86MQVJ30	1635664594476

In the table called “Local Search Query” we can find the text used in the search query in plaintext.

## 4.2.5 Remnants of CL

### /data

## 1. /com.samsung.android.providers.context/databases/ContextLog.db



_id	app_id	app_sub_id	start_time	stop_time
509	com.amazon.avod.thirdpartyclient	com.amazon.avod.watchlist.WatchlistListActivity	2021-10-31 07:31:52.347	2021-10-31 07:31:54.022

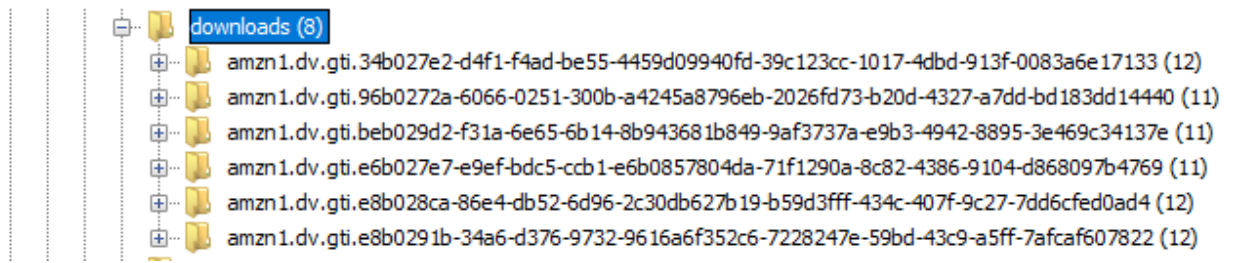
Here is the artifact for adding a video to a list

## 4.2.6 Remnants of DV

### /data

## 1. /com.amazon.avod.thirdpartyclient/files/downloads





Here we can see a list of all the episodes downloaded

File Name	2021-10-31 12:28:39 PKT	2021-10-31 12:28:39 PKT	2021-10-31 12:28:39 PKT	2021-10-31 12:28:39 PKT	71037	Allocated
subtitle_ar-001_SUBTITLE_TYPE_CONTENT.xml						
subtitle_cs-cz_SUBTITLE_TYPE_CONTENT.xml	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	53335	Allocated
subtitle_da-dk_SUBTITLE_TYPE_CONTENT.xml	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	48283	Allocated
subtitle_el-gr_SUBTITLE_TYPE_CONTENT.xml	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	2021-10-31 12:28:37 PKT	65519	Allocated
subtitle_en-us_SDH_TYPE_CONTENT.xml	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	2021-10-31 12:28:25 PKT	78848	Allocated
subtitle_es-419_SUBTITLE_TYPE_CONTENT.xml	2021-10-31 12:28:26 PKT	2021-10-31 12:28:26 PKT	2021-10-31 12:28:26 PKT	2021-10-31 12:28:26 PKT	40462	Allocated

Hex	Text	Application	File Metadata	OS Account	Data Artifacts	Analysis Results	Context	Annotations	Other Occurrences
Strings Indexed Text Translation									
Page: 1 of 5 Page Go to Page: Script: Latin - Basic									
<pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;tt xmlns="http://www.w3.org/ns/ttml" xmlns:tts="http://www.w3.org/ns/ttml#styling" ttp:version="2" xmlns:ttm="http://www.w3.org/ns/ttml#metadata" xmlns:ttp="http://www.w3.org/ns/ttml#parameter" xml:lang="ar"&gt; &lt;head&gt; &lt;styling&gt; &lt;style xml:id="AmazonDefaultStyle" tts:fontFamily="monospace" tts:color="white" tts:fontSize="1c"&gt;&lt;/style&gt; &lt;/styling&gt; &lt;layout&gt; &lt;region xml:id="AmazonDefaultRegion" tts:extent="80% 15%" tts:origin="10% 80%" tts:displayAlign="after" tts:textAlign="center"&gt;&lt;/region&gt; &lt;/layout&gt; &lt;/head&gt; &lt;body&gt; &lt;div&gt; &lt;p begin="00:00:03.629" end="00:00:05.089" region="AmazonDefaultRegion" style="AmazonDefaultStyle"&gt;</pre>									

Inside each of the episode folder, we can find a folder that contains the subtitles for the video in multiple languages.

### 4.2.7 Remnants of CP

No remnants were found for this activity.

### 4.2.8 Remnants of RV

No remnants were found for this activity.

### 4.2.9 Remnants of DP

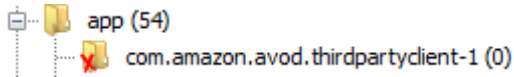
No remnants were found for this activity.

## 4.2.10 Remnants of UA

The Amazon Prime Video application was finally uninstalled from the android mobile device. These artifacts related to the application still remained after it had been uninstalled from the mobile device:

### /app

1. /com.amazon.avod.thirdpartyclient-1



Here we find that the folder has been deleted

### /data

1. /com.amazon.avod.thirdpartyclient

Name	Modified Time	Change Time	Access Time	Created Time	Size
com.amazon.avod.thirdpartyclient	2021-11-01 12:55:29 PKT	2021-11-01 12:55:29 PKT	2021-11-01 12:55:29 PKT	2021-11-01 12:55:29 PKT	1985

This folder has also been deleted

2. /com.google.android.gms/databases/gass.db

Name	Modified Time	Change Time	Access Time	Created Time	Size
gass.db	2021-10-31 11:59:12 PKT	2021-10-31 13:20:00 PKT	2021-10-24 12:43:16 PKT	2021-10-24 12:43:16 PKT	40960

Table	Entries	Page	Export
app_info	60 entries	Page 1 of 1	Export to CSV

_id	pb	package_name	version_code	digest_sha256
59	BLOB Dat...	com.amazon.avod.thirdpartyclient	308015647	7f9c8d08306f9aa8ed8096345e78d0c3f757e67d1dd7023b7944bc72e2666c03

This location still shows information about the amazon video prime application, that it was once downloaded on this mobile device

## 4.2.11 Summary of Remnants

Code of Groups of Activities	Main Directory	Folder	Filename
	1.1)/app	1.1.1)/com.amazon.avod.thirdparty client -1	1.1.1-a) Main Folder
	1.2)/data	1.2.1)/com.amazon.avod.thirdparty	1.2.1-a) Main Folder

1) AI		client 1.2.2) /com.android.vending/databases/ 1.2.3) /com.google.android.gms/databases	1.2.2-a) frosting.db 1.2.2-b) library.db 1.2.2-c) localappstore.db 1.2.3-a) gass.db
	1.3)/user_de	1.3.1) /0/com.amazon.avod.thirdpartyclient	1.3.1-a) Main Folder
2) UL	2.1)/data	2.1.1)/com.samsung.android.providers.context/databases	2.1.1-a) ContextLog.db
3) VV	3.1)/data	3.1.1) /com.samsung.android.providers.context/databases	3.1.1-a) ContextLog.dn
4) SV	4.1)data	4.1.1) /com.samsung.android.providers.context/databases	4.1.1-a) ContextLog.db
		4.1.2) /com.amazon.avod.thirdpartyclient/databases	4.1.2-a) search_query.db
5) CL	5.1)data	5.1.1) /com.samsung.android.providers.context/databases	5.1.1-a) ContextLog.db
6) DV	6.1)data	6.1.1) /com.amazon.avod.thirdpartyclient/files/downloads	6.1.1-a) Episode Folders
7) CP	-	-	-
8) RV	-	-	-
9) DP	-	-	-
10) UA	4.1)/app	4.1.1) /com.amazon.avod.thirdpartyclient-1	4.1.1-a) Main Folder
	4.2)/data	4.2.1) /com.amazon.avod.thirdpartyclient 4.2.2) /com.google.android.gms/databases	4.2.1-a) Main Folder 4.2.2-a) gass.db

Table 4-3 Summary of Amazon Prime Video Remnants with Location Path

### 4.3 iFlix

iFlix is one of the more recent entrants in the market of online video streaming platforms. It was launched in the year 2014 and it primarily catered the Asian market with most of its content originating from China and Korea. As of 2020, iFlix has over 25 million active users and its android application has been downloaded over 50 million times.

Some of the features on this platform are free of cost while for others you have to subscribe to its service. You can view and download the initial episodes for free for most of the TV Shows currently hosted by iFlix, however to for later episodes you need to be a paid subscriber.


The analysis of the android application of iFlix is based on its version 4.6.6.603590720, which is the latest version available on google play store as of now.

### 4.3.1 Remnants of AI

The application can be downloaded and installed from the google play store even if the device is rooted. Following are the remnants discovered after downloading the application, which are grouped by the path at which they were found:

#### /app


1. /iflix.play-1

Name	Modified Time	Change Time	Access Time	Created Time	Size
 iflix.play-1	2021-10-31 11:58:48 PKT	2021-10-31 11:59:08 PKT	2021-10-31 11:56:18 PKT	2021-10-31 11:56:18 PKT	4096

This is the native location for the libraries of all the downloaded applications. The screenshot above shows that the application was downloaded at 11:56:18 AM PKT on 31<sup>st</sup> Oct 2021.

#### /data

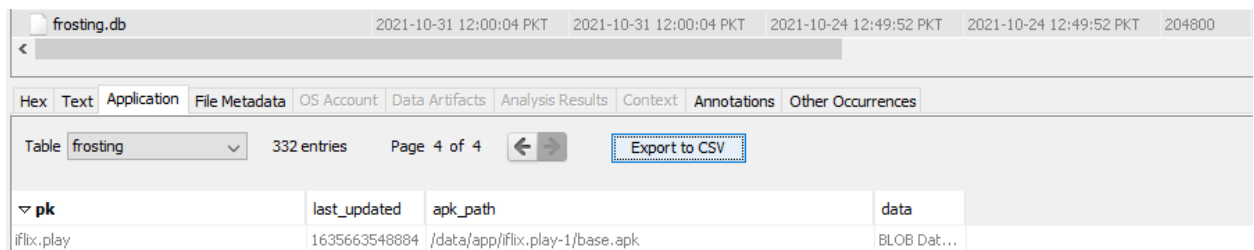
1. /iflix.play

Name	Modified Time	Change Time	Access Time	Created Time	Size
 iflix.play	2021-10-31 13:20:20 PKT	2021-10-31 13:20:20 PKT	2021-10-31 11:59:09 PKT	2021-10-31 11:59:09 PKT	4096

This folder is created soon after the native folder is created in the /app directory. The time of creation of this folder is 11:59:09 AM PKT, the slight delay shows that the folder was first created in the /app directory and then in the /data directory.

2. /com.android.vending/databases

- a) frosting.db



pk	last_updated	apk_path	data
iflix.play	1635663548884	/data/app/iflix.play-1/base.apk	BLOB Dat...

Frosting database contains a table called “frosting” which includes the package name, last updated time, frosting ID and the path to the apk.

b) library.db

The screenshot shows a database viewer interface for 'library.db'. The table 'ownership' is selected, showing 734 entries on page 8 of 8. The table has columns: account, library\_id, backend, doc\_id, doc\_type, offer\_type, documen..., subs\_vali..., app\_cert..., app\_refu..., app\_refu..., and subs\_aut. The first row of data is: adilahmad95@gmail.com, 3, 3, iflix.play, 1, 1, -8124028..., e2sikODL..., 0, 0.

account	library_id	backend	doc_id	doc_type	offer_type	documen...	subs_vali...	app_cert...	app_refu...	app_refu...	subs_aut...
adilahmad95@gmail.com	3	3	iflix.play	1	1	-8124028...		e2sikODL...	0	0	

The table called “ownership” contains information about all the packages downloaded from google play store along with the account used to download them.

c) localappstore.db

The screenshot shows a database viewer interface for 'localappstate.db'. The table 'appstate' is selected, showing 42 entries on page 1 of 1. The table has columns: package\_name, auto\_up..., desired..., downloa..., delivery\_data, delivery\_data..., installer..., first\_download..., referrer, and account. The first row of data is: iflix.play, 1, -1, BLOB Data not shown, 1635663380848, 0, 1635663420297, adilahmad95@gmail.com.

package_name	auto_up...	desired_...	downloa...	delivery_data	delivery_data...	installer_...	first_download...	referrer	account
iflix.play	1	-1		BLOB Data not shown	1635663380848	0	1635663420297		adilahmad95@gmail.com

The table called “appstore” also shows information about the packages downloaded from google play store.

3. /com.google.android.gms/databases/gass.db


The screenshot shows a database viewer interface for 'gass.db'. The table 'app\_info' is selected, showing 60 entries on page 1 of 1. The table has columns: \_id, pb, package\_name, version\_code, and digest\_sha256. The first row of data is: 60, BLOB Dat..., iflix.play, 603590720, 975d094ba482cd1b0aa903f1f65f7822a7901252fb1f3f96013091c7dc3bb9f5.

_id	pb	package_name	version_code	digest_sha256
60	BLOB Dat...	iflix.play	603590720	975d094ba482cd1b0aa903f1f65f7822a7901252fb1f3f96013091c7dc3bb9f5

In the Gass database, there is a table called “app\_info” which contains the information about the installed android applications. The iFlix package along with its package name, version and package hash were found. The hashing algorithm used to determine the package hash is SHA-256.

## /user\_de

1. /0/iflix.play

Name	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)
 iflix.play	2021-10-31 12:00:05 PKT	2021-10-31 12:00:05 PKT	2021-10-31 11:59:09 PKT	2021-10-31 11:59:09 PKT	4096	Allocated

At this location we found a folder with the package name of iFlix and its creation time corresponds with the installation of the application.

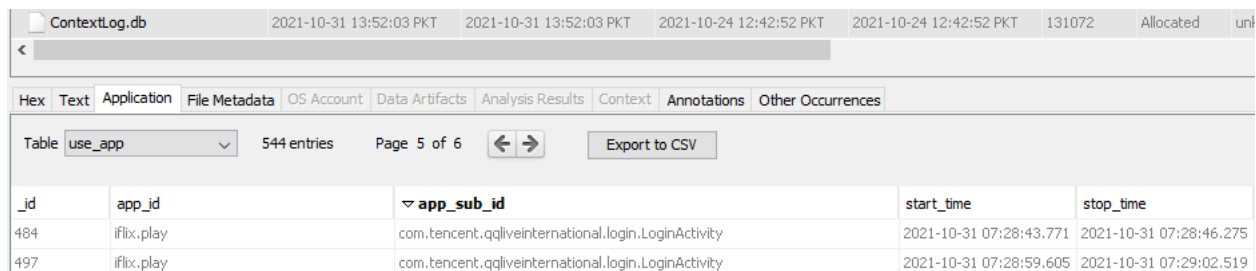
### 4.3.2 Remnants of UL

You can sign up and subscribe using the android application of iFlix and if you are already a subscriber, you can just enter your credentials and login to the application. Furthermore, as stated earlier that in order to just use the basic functions of the application you are not required to be a paid subscriber.

Following are the remnants discovered while logging in to iFlix application:

## /data

1. /com.samsung.android.providers.context/databases/ContextLog.db



The screenshot shows a forensic tool interface with a table of data. The table has columns for \_id, app\_id, app\_sub\_id, start\_time, and stop\_time. The data rows show login activity for the iflix.play application.

_id	app_id	app_sub_id	start_time	stop_time
484	iflix.play	com.tencent.qqliveinternational.login.LoginActivity	2021-10-31 07:28:43.771	2021-10-31 07:28:46.275
497	iflix.play	com.tencent.qqliveinternational.login.LoginActivity	2021-10-31 07:28:59.605	2021-10-31 07:29:02.519

In the table called “use app” we can view the artifacts from the login activity along with the timestamps.

### 4.3.3 Remnants of VV

## /data

1. /iflix.play/databases/videointernational.db

id	vid	cid	user_id	modify_time	video_play_time	video_total_time	full_episode_count	update_...	update_t...	episode	publi
1	v0040pr89t9	air11ooo2rdsd3		1635665315827	1024	2835	30	30	0	1	0
2	n0040dxjdal	d33pxzjcu77yujn	4181220356	1635665431072	578	2473	24	22	0	1	0
3	l0040latjit	d33pxzjcu77yujn	4181220356	1635665651666	6	2794	24	22	0	17	0
4	v0040pr89t9	air11ooo2rdsd3	4181220356	1635665581705	1	2835	30	30	0	1	0

In the table called “t\_history\_record” we can see entries for every time a video was played along with its playtime and the total duration of the video.

### 4.3.4 Remnants of SV

/data

1. /com.samsung.android.providers.context/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
518	iflix.play	com.tencent.qqlive118n.search.SearchActivity	2021-10-31 07:33:06.322	2021-10-31 07:33:25.640

In the table called “use\_app”, we can see the artifact for a search activity.

### 4.3.5 Remnants of CL

No remnants were found for this activity.

### 4.3.6 Remnants of DV

/data

1. /com.samsung.android.providers.context/databases/ContextLog.db

_id	app_id	app_sub_id	start_time	stop_time
535	iflix.play	com.tencent.qqliveinternational.download.video.downloading.DownloadingActivity	2021-10-31 07:35:02.406	2021-10-31 07:35:12.465
536	iflix.play	com.tencent.qqliveinternational.download.video.downloading.DownloadingActivity	2021-10-31 07:40:57.958	2021-10-31 07:41:29.399
537	iflix.play	com.tencent.qqliveinternational.download.video.downloading.DownloadingActivity	2021-10-31 08:19:59.526	2021-10-31 08:19:59.603
539	iflix.play	com.tencent.qqliveinternational.download.video.downloading.DownloadingActivity	2021-10-31 08:20:01.782	2021-10-31 08:20:06.258

Here we can see the artifacts from an activity where a video was downloaded

## 2. /iflix.play/files

record_id	vid	format	data	state	charge	errcode	last_mod...	ext
v0040pr89t9.ld	v0040pr89t9	ld	<?xml version="1.0" encoding="UTF-8"?><Table cover_id="" video_name="" image_url="" module_i... 3	3	0	0	0	

In this database there is a table called “download\_record” which shows the records of all videos downloaded

### 4.3.7 Remnants of CP

No remnants were found for this activity.

### 4.3.8 Remnants of RV

No remnants were found for this activity.

### 4.3.9 Remnants of DP

No remnants were found for this activity.

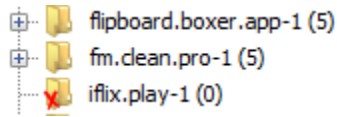
### 4.3.10 Remnants of UA

The iFlix application was finally uninstalled from the android mobile device. These artifacts related to the application still remained after it had been uninstalled from the mobile device:



## /app


1. /iflix.play-1



Here we find a deleted folder with the package name

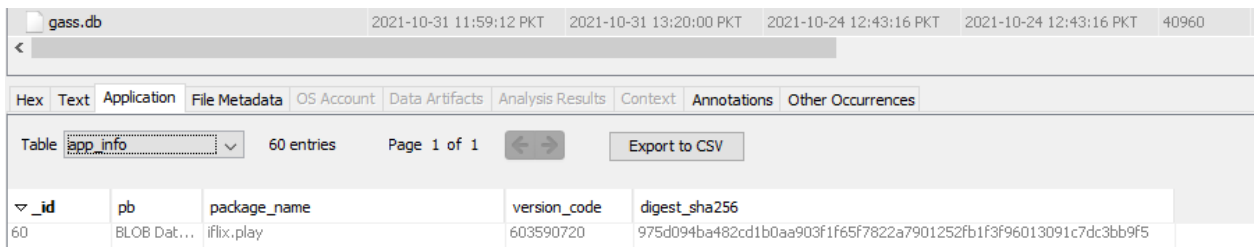
## /data

1. /iflix.play

Name	Modified Time	Change Time	Access Time	Created Time	Size
 iflix.play	2021-11-01 12:50:24 PKT	2021-11-01 12:50:24 PKT	2021-11-01 12:49:32 PKT	2021-11-01 12:49:32 PKT	0

This folder has also been deleted

2. /com.google.android.gms/databases/gass.db



_id	pb	package_name	version_code	digest_sha256
60	BLOB Dat...	iflix.play	603590720	975d094ba482cd1b0aa903f1f65f7822a7901252fb1f3f96013091c7dc3bb9f5

This location shows that the application was once downloaded on the android mobile device.

### 4.3.11 Summary of Remnants

Code of Groups of Activities	Main Directory	Folder	Filename
1) AI	1.1)/app	1.1.1)/iflix.play-1	1.1.1-a) Main Folder
	1.2)/data	1.2.1)/iflix.play	1.2.1-a) Main Folder
		1.2.2)	1.2.2-a) frosting.db
		/com.android.vending/databases/	1.2.2-b) library.db
1.2.3)	/com.google.android.gms/databases	1.2.2-c) localappstore.db	
		1.2.3-a) gass.db	
1.3)/user_de	1.3.1) /0/iflix.play	1.3.1-a) Main Folder	
2) UL	2.1)/data	2.1.1)/com.samsung.android.providers.contexty/databases	2.1.1-a) ContextLog.db

3) VV	3.1)/data	3.1.1)/iflix.play/databases	3.1.1-a) videointernational.db
4) SV	4.1)/data	4.1.1) /com.samsung.android.providers.context/databases	4.1.1-a) ContextLog.db
5) CL	-	-	-
6) DV	6.1)/data	6.1.1) /com.samsung.android.providers.context/databases 6.1.2)/iflix.play/files	6.1.1-a) ContextLog.db 6.1.2-a) Downloaded Episodes
7) CP	-	-	-
8) RV	-	-	-
9) DP	-	-	-
10) UA	4.1)/app	4.1.1) /iflix.play-1	4.1.1-a) Main Folder
	4.2)/data	4.2.1) /iflix.play 4.2.2) /com.google.android.gms/databases	4.2.1-a) Main Folder 4.2.2-a) gass.db

*Table 4-4 Summary of iFlix Remnants with Location Path*

## Chapter – 5

### 5 Results & Discussions

Following our extensive study and examination of the three most popular android video streaming applications, we were able to collect and present vital remnants from the applications with significant forensic importance. With the help of these remnants, we were able to answer the research questions mentioned in the earlier chapters. These questions were as follows:

- What is the nature of the remnants left behind corresponding to different activities by the three android video streaming applications and how much forensic value do they hold?
- How do the remnants left behind by the three applications compare with one another?

Now let's take a look at the details of the artifacts with respect to the different activities performed in the applications.

#### 1) Application Installation

Following artifacts were discovered after the installation phase of the application:

- Timestamp at which the application was downloaded
- Application Title
- Application Package Name
- Application Hash
- Application Version
- The Google Account used to Download the Application
- Application Last Update Time
- Application Path

These artifacts prove very useful from a forensics point of view. The application hash can be used to verify the integrity of the application. Furthermore, the application version informs the investigator about the features that the application has. The Google account used to download the application provides information about the suspect.

#### 2) User Login

Following artifacts were discovered after the user logged into the application:

- The Timestamp at which the User Logged In
- The Application Name in to which he Logged In

The artifacts corresponding to user login activity provide information to the forensics investigator about the time at which the user logged in and into which application.

### **3) Viewing a Video**

Following artifacts were discovered after the user viewed a video on the application:

- The Time at which the Video Started Playing
- The Time at which the Video Stopped Playing
- The Name of the Application

In fact, this is the most crucial artifact as it allows the forensics investigator to map the timeline of the activities of the suspect and use that to prove his involvement in any criminal activity.

### **4) Searching for a Video**

Following artifacts were discovered after the user searched for a video on the application:

- The Starting Time of the Activity
- The Finishing Time of the Activity
- The Name of the Application
- The Search Query Entered in the Search Bar

These artifacts can inform the forensics investigator about the likes and dislikes of the suspect based on their search queries.

### **5) Creating a List**

Following artifacts were discovered after the user created a list of videos on the application:

- The Time at which the Activity Started
- The Time at which the Activity Stopped
- The Name of the Application

These artifacts also inform the forensics investigator about the preferences of the suspect.

### **6) Downloading a Video**

Following artifacts were discovered after the user downloaded a video in order to view it offline later on the application:

- Title of the Episode
- Title of the TV Show
- Poster of the Episode
- Poster of the TV Show
- Subtitle Files in Various Languages
- The Timestamp at Which the Video was Downloaded

### 7) Creating User Profiles

No artifacts were discovered for this activity

### 8) Rating a Video

No artifacts were discovered for this activity

### 9) Deleting a User Profile

No artifacts were discovered for this activity

### 10) Uninstalling the Application

Following artifacts were still available even the application had been uninstalled from the android mobile device:

- Application Title
- Application Package Name
- Application Hash
- Application Version
- Some User Activities

These artifacts prevent the suspect from denying any activity by deleting the application. The forensics investigator will still be to discover the suspect's activity even after the suspect has deleted the application.

## 5.1 Netflix Analysis Summary

Code of Groups of Activities	Artifacts	Location
AI	Application Download Time	/app/com.netflix.mediaclient-1
	Application Title	/app/com.netflix.mediaclient-1
	Application Package Name	/app/com.netflix.mediaclient-1

	Application Hash	/data/com.google.android.gms/databases/gass.db
	Application Version	/data/com.google.android.gms/databases/google_app_measurement.db
	Application Last Update Time	/app/com.netflix.mediaclient-1
	Application Path	/data/com.android.vending/databases/frosting.db
UL	Login Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.context/databases/ContextLog.db
VV	Video Start Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Video Stop Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.context/databases/ContextLog.db
SV	Activity Start Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Activity End Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Name of the Application	/data/com.samsung.android.providers.context/databases/ContextLog.db
CL	-	-
DV	Title of the Episode	/data/com.netflix.mediaclient/databases/OfflineDb
	Title of the TV Show	/data/com.netflix.mediaclient/databases/OfflineDb
	Poster of the Episode	/data/com.netflix.mediaclient/file/img/of/videos
	Poster of the TV Show	/data/com.netflix.mediaclient/file/img/of/videos
CP	-	-
RV	-	-
DP	-	-
UA	Application Title	/app/com.netflix.mediaclient-1
	Application Package Name	/app/com.netflix.mediaclient-1
	Application Hash	/data/com.google.android.gms/databases/gass.db
	Application Version	/data/com.google.android.gms/databases/google_app_measurement.db
	Some User Activities	/data/com.samsung.android.providers.context/databases/ContextLog.db

*Table 5-1 Summary of iFlix Remnants with Location Path Summary of Netflix Artifacts*

## 5.2 Amazon Prime Video Analysis Summary

Code of Groups of Activities	Artifacts	Location
AI	Application Download Time	/app/com.amazon.avod.thirdpartyclient-1
	Application Title	/app/com.amazon.avod.thirdpartyclient-1
	Application Package Name	/app/com.amazon.avod.thirdpartyclient-1
	Application Hash	/data/com.google.android.gms/databases/gass.db
	Application Version	/data/com.google.android.gms/databases/gass.db
	Google Account	/data/com.android.vending/databases/library.db
	Application Last Update Time	/app/com.amazon.avod.thirdpartyclient-1
	Application Path	/data/com.android.vending/databases/frosting.db
UL	Login Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.context/databases/ContextLog.db
VV	Video Start Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Video Stop Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.context/databases/ContextLog.db
SV	Activity Start Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Activity End Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Name of the Application	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Search Query	/data/com.amazon.avod.thirdpartyclient/databases/search_query.db
CL	Activity Start Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Activity Stop Time	/data/com.samsung.android.providers.context/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.context/databases/ContextLog.db
DV	Subtitle Files	/com.amazon.avod.thirdpartyclient/files/downloads
	Timestamp of Download	/com.amazon.avod.thirdpartyclient/files/downloads
CP	-	-
RV	-	-

DP	-	-
UA	Application Title	/app/com.amazon.avod.thirdpartyclient-1
	Application Package Name	/app/com.amazon.avod.thirdpartyclient-1
	Application Hash	/data/com.google.android.gms/data bases/gass.db
	Application Version	/data/com.google.android.gms/data bases/gass.db
	Some User Activities	/data/com.samsung.android.providers.con text/databases/ContextLog.db

Table 5-2 Summary of iFlix Remnants with Location Path Summary of Amazon Prime Video Artifacts

### 5.3 iFlix Analysis Summary

Code of Groups of Activities	Artifacts	Location
AI	Application Download Time	/app/iflix.play-1
	Application Title	/app/iflix.play-1
	Application Package Name	/app/iflix.play-1
	Application Hash	/data/com.google.android.gms/data bases/gass.db
	Application Version	/data/com.google.android.gms/data bases/gass.db
	Google Account	/data/com.android.vending/databases/ library.db
	Application Last Update Time	/app/iflix.play-1
UL	Login Time	/data/com.samsung.android.providers.con text/databases/ContextLog.db
	Application Name	/data/com.samsung.android.providers.con text/databases/ContextLog.db
VV	Video Start Time	/data/iflix.play/databases/videointernation al.db
	Video Stop Time	/data/iflix.play/databases/videointernation al.db
	Application Name	/data/iflix.play/databases/videointernation al.db
SV	Activity Start Time	/data/com.samsung.android.providers.con text/databases/ContextLog.db
	Activity End Time	/data/com.samsung.android.providers.con text/databases/ContextLog.db
	Name of the Application	/data/com.samsung.android.providers.con text/databases/ContextLog.db
CL	-	-
DV	Timestamp of Download	/data/com.samsung.android.providers.con text/databases/ContextLog.db
CP	-	-



RV	-	-
DP	-	-
UA	Application Title	/app/iflix.play-1
	Application Package Name	/app/iflix.play-1
	Application Hash	/data/com.google.android.gms/databases/gass.db
	Application Version	/data/com.google.android.gms/databases/gass.db
	Some User Activities	/data/com.samsung.android.providers.context/databases/ContextLog.db

Table 5-3 Summary of iFlix Remnants with Location Path Summary of iFlix Artifacts

## 5.4 Comparison of Available Artifacts

Code of Groups of Activities	Artifacts	Netflix	Amazon Prime Video	iFlix
AI	Application Download Time	Yes	Yes	Yes
	Application Title	Yes	Yes	Yes
	Application Package Name	Yes	Yes	Yes
	Application Hash	Yes	Yes	Yes
	Application Version	Yes	Yes	Yes
	Google Account	No	Yes	Yes
	Application Last Update Time	Yes	Yes	Yes
	Application Path	Yes	Yes	Yes
UL	Login Time	Yes	Yes	Yes
	Application Name	Yes	Yes	Yes
VV	Video Start Time	Yes	Yes	Yes
	Video Stop Time	Yes	Yes	Yes
	Application Name	Yes	Yes	Yes
SV	Activity Start Time	Yes	Yes	Yes
	Activity End Time	Yes	Yes	Yes
	Name of the Application	Yes	Yes	Yes
	Search Query	No	Yes	No
CL	Activity Start Time	No	Yes	No
	Activity Stop Time	No	Yes	No
	Application Name	No	Yes	No
DV	Title of the Episode	Yes	No	No
	Title of the TV Show	Yes	No	No
	Poster of the Episode	Yes	No	No
	Poster of the TV Show	Yes	No	No
	Subtitle Files	No	Yes	No
	Timestamp of Download	No	Yes	Yes
CP	-	No	No	No

RV	-	No	No	No
DP	-	No	No	No
UA	Application Title	Yes	Yes	Yes
	Application Package Name	Yes	Yes	Yes
	Application Hash	Yes	Yes	Yes
	Application Version	Yes	Yes	Yes
	Some User Activities	Yes	Yes	Yes

*Table 5-5 Summary of iFlix Remnants with Location Path Comparison of Artifacts*

## Chapter - 6

### 6 Conclusion and Future Horizons

Online video streaming platforms have gained popularity globally in the recent past. This can be credited to the fact that online video streaming platforms provide an easier alternative to consuming entertainment content as compared to traditional electronic media. This gain in popularity has been further amplified by a recent pandemic of the Corona Virus where we saw cinemas closing down for indefinite periods. In these times, online video streaming platforms were the only source of entertainment for people. With all these benefits that come along with online video streaming platforms, there are some drawbacks as well. These include the potential risk of privacy caused when mobile devices store personal information and the history of user activities performed while interacting with the mobile applications of these video streaming platforms. Looking at these stored artifacts from a different angle, we come to find out that they can also prove helpful in a forensics investigation especially where we need to validate the alibi of the suspect. The timeline of activities confessed by a suspect can be verified against the activity log stored behind by the video streaming applications.

After an extensive literature review, we came to know that a forensic study of such nature has never been conducted on our category of android applications. Therefore, we went ahead with this research project where we targeted two very important questions. The first one being that what artifacts are left behind by the android video streaming applications along with their location. Secondly, we tried to answer the question that how do these artifacts collected from different applications compare with each other.

In our study, we chose the three most popular android video streaming applications as part of our target applications. These applications were Netflix, Amazon Prime Video, and iFlix. These applications have been downloaded several million times from the Google Play Store and can be considered the most popular. Our examination and analysis of the artifacts left behind the applications were related to the installation, login, viewing videos, searching for videos, downloading videos, creating a list of videos, etc.

Extracted results included several artifacts with great forensics value. These contained the account information of the user who downloaded the application, information about the application such as its package name and version. More importantly, we could exactly state the timestamps between which a user was viewing a video. This helps in reliably formulating the timeline of activities of the suspect should he be involved in any criminal activity such as a car accident. Furthermore, artifacts related to downloading a video were also found.

The artifacts, apart from having forensic value, also pose a privacy risk. As so much information related to the user is stored on the mobile device, it can prove detrimental to the user should the mobile device be accessed by malicious actors. Mobile application developers should limit the amount of information stored on mobile devices and try to encrypt it as much as possible.

## **6.1 Future Work**

In this research, we tried to be as comprehensive as possible given our constraints. However, due to limitations of time and technology, there still remains potential for future work in this domain. The focus of our study was on the physical image of the device storage and in the future people can work on acquiring the volatile memory when an application is running to study the artifacts present in it. Secondly, we carried out research on android based applications as this is the more popular mobile OS, however, a similar study can be conducted on iOS-based video streaming applications and the artifacts can be compared with those collected from android applications to formulate a comparative study. Finally, this study was conducted on the most popular video streaming platforms globally and in the future researchers can focus on the local alternatives of these applications.

## 7 References

- [1] M. Sudozai, S. Saleem and W. J. Buchanan, "Forensics study of IMO call and chat app," *Digital Investigation*, pp. 1-19, 2018.
- [2] J. Gregorio, A. Gardel and B. Alarcos, "Forensic analysis of Telegram Messenger for Windows Phone," *Digital Investigation*, vol. 22, pp. 88-106, 2017.
- [3] S. Knox, S. Moghadam and K. Patrick, "What's really 'Happn ing'? A forensic analysis of Android and iOS Happn dating apps," *Computer Secur.*, vol. 94, 2020.
- [4] A. Afzal, M. Hussain and S. Saleem, "Encrypted Network Traffic Analysis of Secure Instant Messaging Application: A Case Study of Signal Messenger App," *MDPI*, 2021.
- [5] G. He, B. Xu and H. Zhu, "Identifying Mobile Applications for Encrypted Network Traffic," *CBD*, 2017.
- [6] D. Walnycky, I. Baggili and A. Marrington, "Network and device forensic analysis of Android social-messaging applications," *Digital Investigation*, vol. 14, pp. 77-84, 2015.
- [7] R. Umar, I. Riadi and B. F. Muthohirin, "Live forensics of tools on android devices for email forensics," *Telkomnika*, vol. 17, no. 4, pp. 1803-1809, 2019.
- [8] A. Fauzan, I. Riadi and A. Fadlil, "Analisis Forensik Digital Pada Line Messenger Untuk," *Annual Research Seminar*, vol. 2, no. 1, 2016.
- [9] N. Anwar and I. Riadi, "Analisis Investigasi Forensik WhatsApp Messenger Smartphone Terhadap WhatsApp Berbasis Web," *JITEKI*, vol. 3, no. 1, 2017.
- [10] N. A. Mutawa, I. Baggili and A. Marington, "Forensic analysis of social networking applications on mobile devices," *Digital Investigation*, vol. 9, pp. 24-33, 2012.
- [11] F. A. Awan, "Forensic examination of social networking applications on smartphones," *CIACS*, 2015.
- [12] H. Zhang, L. Chen and Q. Liu, "Digital Forensic Analysis of Instant Messaging Applications on Android Smartphones," *ICNC*, 2018.
- [13] H. Nurhairani and I. Riadi, "Analysis Mobile Forensics on Twitter Application using the National Institute of Justice (NIJ) Method," *International Journal of Computer Applications*, 2019.
- [14] R. Saputra and I. Riadi, "Forensic Browser of Twitter based on Web Services," *IJCA*, vol. 175, no. 29, pp. 34-39, 2020.
- [15] M. R. Arshad, M. Hussain and H. Tahir, "Forensic Analysis of Tor Browser on Windows 10 and Android 10 Operating Systems," *IEEE Access*, 2021.
- [16] G. M. Zamroni and I. Riadi, "Instant Messaging Forensic Tools Comparison on Android Operating

- System," *Kinetik*, vol. 4, pp. 137-148, 2019.
- [17] I. Riadi and G. M. Zamroni, "Mobile Forensic Tools Evaluation for Digital Crime Investigation," *International Journal on Advanced Science Engineering and Information Technology*, 2018.
- [18] G. Horsman, "Tool testing and reliability Issues in the field of digital forensics," *Digital Investigation*, 2019.
- [19] M. A. Talib, "Testing closed source software: computer forensic tool case study," *Journal of Computer Virology and Hacking Techniques*, vol. 14, pp. 167-179, 2018.
- [20] T. Wu, F. Breitingner and S. O'Shaughnessy, "Digital forensic tools: Recent advances and enhancing the status quo," *Digital Investigation*, vol. 34, 2020.
- [21] H. H. Lwin, W. P. Aung and K. K. Lin, "Comparitive Analysis of Android Mobile Forensics," *IEEE Conference on Computer Applications*, 2020.
- [22] P. Fend, Q. Li and P. Z. e. al., "Logical acquisition method based on data migration for Android mobile devices," *Digital Investigation*, vol. 26, pp. 55-62, 2018.
- [23] A. Fukami, R. Stoykova and Z. Geradts, "A new model for forensic data extraction from encrypted mobile devices," *Digital Investigation*, vol. 38, 2021.
- [24] C. Anglano, M. Canonico and M. Guazzone, "The Android Forensics Automator (AnForA): A tool for the Automatic Forensic Analysis of Android Applications," *Computers and Security*, vol. 88, 2020.
- [25] Q. Luo, J. Liu and J. Wang, "Automatic Content Inspection and Forensics for Children Android Apps," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7123-7134, 2020.
- [26] S. Krishnan, B. Zhou and M. K. An, "Smartphone Forensic Challenges," *International Journal of Computer Science and Security*, vol. 13, no. 5, 2019.
- [27] B. Liu, Y.-x. Chang and L.-r. Cong, "A Memory Acquisition Method for Android Application Forensics," *Journal of Physics*, vol. 1314, 2019.
- [28] X. Zhang, F. Breitingner and E. Luechinger, "Android Application forensics: A survey of obfuscation, obfuscation detection and deobfuscation techniques and their impact on investigations," *Digital Investigation*, vol. 39, 2021.
- [29] N. Sunde and I. E. Dror, "Cognitive and human factors in digital forensics: Problems, challenges and the way forward," *Digital Investigation*, vol. 29, pp. 101-108, 2019.
- [30] L. A. Herrera, "Challenges of acquiring mobile devices while minimizing the loss of usable forensics data," *Internation Symposium on Digital Forensics*, 2020.