Effects of Timestamp Manipulation in a Car Audio Video Navigation System Connected to a Smartphone via Bluetooth: A Preliminary Study

Minhyuk Cho
Dept. of Mobile System Engineering
Dankook University
Yongin, Republic of Korea
cgumgek8@dankook.ac.kr

Sunjae Kim
Dept. of Computer Science and Engineering
Dankook University
Yongin, Republic of Korea
rlatjswo0824@dankook.ac.kr

Seong-je Cho
Dept. of Software Science
Dankook University
Yongin, Republic of Korea
sjcho@dankook.ac.kr

***Abstract*— In digital forensics, timestamps are crucial for analyzing key events in chronological order, making them vital to forensic investigations. This paper presents a method for detecting attempts to conceal a crime through the manipulation of timestamps on IT devices. Specifically, we introduce a technique to identify timestamp manipulation in a car's infotainment system when the suspect's smartphone is connected via Bluetooth. The method involves extracting and analyzing Bluetooth Host Controller Interface (HCI) logs, Bluetooth logs, and system logs from both the smartphone and the infotainment system. Scenario-based experiments demonstrate the effectiveness of this technique in detecting timestamp manipulation on the infotainment system.**

Keywords—Digital Forensic, Audio Video Navigation, Timestamp manipulation, Log analysis, Anti-forensic

I. Introduction

Digital forensics is a process that involves the scientific collection and analysis of data stored in IT devices such as computers and smartphones to resolve criminal cases [1]. In digital forensics, timestamps are considered crucial for reconstructing events in chronological order and are essential for accurately analyzing the time of events [2]. For this reason, malicious users or criminals may manipulate the timestamps on IT devices to conceal their actions.

Since timestamp manipulation attacks can tamper with evidence and hinder digital forensic investigations, research aimed at detecting timestamp manipulation is essential. In response, some researchers have proposed methods to detect timestamp manipulation that may occur within file systems [3][4].

Meanwhile, as in-vehicle infotainment (IVI) systems connect with drivers' smartphones via Bluetooth to provide various convenient services, these systems store valuable data that can be used in digital forensic investigations. Consequently, IVI systems have become a focus of digital forensic studies [5][6][7]. However, research on detecting timestamp manipulation in IVI systems connected to smartphones via Bluetooth remains insufficient. Note that an IVI system is also called a car audio video navigation (AVN) system.

This paper proposes a novel technique for detecting timestamp manipulation in an Android-based AVN system connected to a driver's smartphone via Bluetooth. In this work, we manipulate only the time on the AVN system while leaving the smartphone's time untouched and then trigger several events depending on a specific scenario. Next, we collect various logs (HCI snoop logs, Bluetooth logs, system logs) from the AVN system as well as the smartphone, and carefully analyze the collected logs, where HCI refers to the Host Controller Interface. Through log collection and analysis, we demonstrate that the proposed technique can effectively determine which device's timestamp was manipulated.

II. Related Work

Oh et al. [8] reviewed various existing studies on detecting timestamp manipulation in NTFS file systems. Their research concluded that journal-based detection methods are the most effective; however, they identified a limitation in that traditional journal-based detection methods are difficult to apply in actual digital forensic investigations. To address this issue, they conducted further research to propose a new detection algorithm, and experiments demonstrated that the proposed algorithm outperforms existing methods in performance.

Kaart et al. [9] studied detecting time and date manipulation in Android forensics. They emphasized the importance of validation using reference devices that share the same brand, model, and Android version to address issues related to time and date manipulation. In addition, they proposed a straightforward method for time manipulation and analyzed the settings related to time synchronization and their potential impact.

Pieterse et al. [10] proposed the Authenticity Framework for Android Timestamps (AFAT) to detect timestamp manipulation on Android devices. This framework is designed to verify the integrity of timestamps, which play a crucial role in digital forensic investigations. AFAT employs two primary methods to determine whether timestamps have been manipulated. The first method identifies specific changes in the Android file system, which indicate potential tampering of the SQLite database's integrity. The second method analyzes inconsistencies within the SQLite database itself to detect whether timestamps have been altered. These methods are vital in verifying the authenticity of timestamps

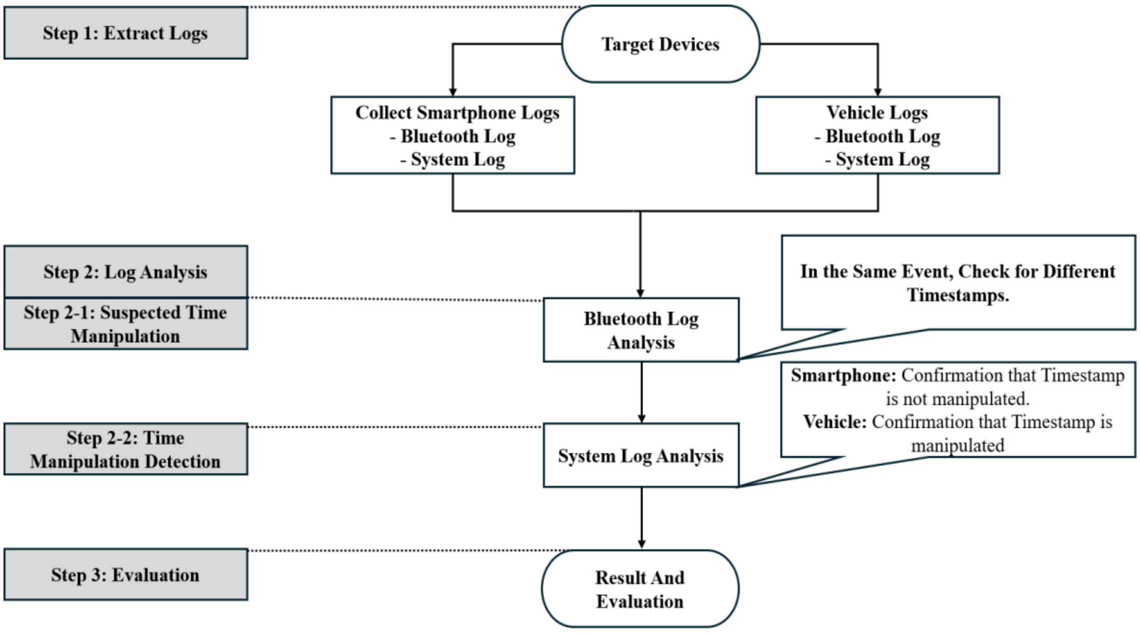
Step 1: Extract Logs
Target Devices
Collect Smartphone Logs
- Bluetooth Log
- System Log
Vehicle Logs
- Bluetooth Log
- System Log
Step 2: Log Analysis
Step 2-1: Suspected Time Manipulation
Bluetooth Log Analysis
In the Same Event, Check for Different Timestamps.
Smartphone: Confirmation that Timestamp is not manipulated.
Vehicle: Confirmation that Timestamp is manipulated
Step 2-2: Time Manipulation Detection
System Log Analysis
Step 3: Evaluation
Result And Evaluation

Fig 1. Process of Detecting Timestamp Manipulation on an AVN system

and identifying manipulated evidence in digital forensic investigations.

In particular, Pieterse's research emphasizes the importance of understanding the chronological sequence of events based on various logs and timestamps generated by Android smartphones, which is essential for the success of forensic investigations. The study introduces concrete methodologies for detecting attempts by smartphone users or malicious applications to manipulate timestamps and obstruct investigations. This research contributes significantly to countering anti-forensic techniques, offering tools to prevent evidence distortion caused by timestamp manipulation.

III. Timestamp Manipulation Detection Technique on an AVN System

The process for the proposed timestamp manipulation detection technique is illustrated in **Figure 1**. First, in an environment where a car's AVN system and the driver's Android smartphone are connected via Bluetooth, we design an event-triggering scenario. We only manipulated the AVN's timestamp and generated some events according to the scenario. Next, log data is collected from both an Android phone and a car AVN system using the methods presented in **Table 1**.

TABLE 1. Method for collecting logs

Android smartphone	*Vehicle's AVN*
Logcat	Copy Logs to USB
Bugreport	Chip-off

For the Android phone, logs are acquired using Android Debug Bridge (ADB) commands such as **'logcat'** or **'bugreport'.** The '**logcat**' command only extracts volatile logs, which are lost when the device is powered off. On the other hand, the '**bugreport'** command retrieves comprehensive system log data, including volatile logs [11]. Therefore, in this paper, we use the '**bugreport**' command to extract logs from the Android phone.

One challenge is accessing logs generated by the Bluetooth app on the Android phone, which typically requires the device to be rooted. As another solution, enabling the 'Bluetooth HCI Snoop Log' feature on the Android phone allows for the easy collection of Bluetooth logs without the need to root the device.

For the car AVN system, logs can be retrieved using either the chip-off method or Engineering Mode. The **'chip-off'** method physically removes the memory chip from the vehicle to extract data, allowing for the collection of extensive log data, however this method risks damaging the AVN system [12]. In contrast, the Engineering Mode of the AVN system is a hidden feature that provides a menu for logically extracting logs. In this study, we use the **'USB Copy'** menu provided by the Engineering Mode to extract logs from the AVN system.

The next step involves analyzing the extracted logs. We first analyze the Bluetooth logs to determine whether any timestamp was manipulated. If the Bluetooth logs capture any sign that any timestamp has been tampered, we proceed to analyze the system logs to confirm whether which timestamp has been tampered. Through system log analysis, we can identify which device's timestamp has been manipulated and determine the accurate time information.

IV. Experimental Environment and Method

A. Experimental Environment

We conducted experiments on the AVN system installed in a Hyundai Avante vehicle, which runs Android 4.4.2 KitKat, and a Galaxy S21 smartphone running Android version 14. The detailed specifications of the target devices are listed in **Table 2**.

TABLE 2. SPECIFICATION OF TARGET AVN SYSTEM AND ANDROID PHONE

Car AVN	
Vehicle Model	Hyundai Avante
AVN Manufacturer	Hyundai Mobis
Operating System	Android 4.4.2(KitKat)
Kernel Version	Linux 3.18.24-tcc
Smartphone	
Mobile Device Model	Galaxy S21
Manufacturer	Samsung
Operating System	Android 14
Kernel Version	Linux 5.4.242-27760517-abG991NKSU4FWK7

B. Scenario-based Event Generation

In this study, we generated log data by conducting experiments based on an event-driven scenario. Specifically, the scenario includes several events or actions that a driver might perform while driving, as shown in **Table 3**. Those events were triggered based on the scenario.

TABLE 3. EVENT SCENARIO

Time	Event	Description
15:23	Bluetooth Connect	Enabling “Bluetooth HCI Snoop Log” on Android smartphone
15:24	Network Time Off	Turn off only vehicle’s network time
15:41	Manipulate Vehicle Time	2024/09/05, 15:41 -> 2024/08/25, 13:40
15:41	Calling Event	Calling to ‘01049232198’
15:43	Music Event	Title: A Collection of 2000s hit
15:44	Log Dump	Smartphone: adb bugreport Vehicle: Copy Image to USB

C. Extracting Log Files in Androidphone And AVN system

After triggering events based on the designed scenario, we extracted log files from both the Android phone and the AVN system. Among the extracted log files, we selected those related to modified date tags and timestamps. The log files selected for forensic investigation are listed in **Table 4**.

TABLE 4. TARGET LOG FILES FOR FORENSIC ANALYSIS

Smartphone	*Note*	*Vehicle*	*Note*
btsnoop_hci.log	Bluetooth Packet File	bluetooth Log	Bluetooth Folder
dumpstate-2024-09-05-15-43-02.txt	System Diagnostic Log Files	telematics.txt	Log files related to the vehicle's telematics system
		SET_USER_TIME@1724820000044.txt	Log files related to time-related operations

V. ANALYSYS OF BLUETOOTH LOG AND SYSTEM LOG

Since the Bluetooth logs extracted from the smartphone stored in a packet form, we used the packet analysis tool WireShark [13] to analyze the extracted logs. For analyzing the system logs stored in text format, we utilized tools such as Autopsy, Notepad, and VSCode.

A. Bluetooth Log

Analysis of the Bluetooth logs extracted from both the smartphone and the AVN system revealed that different timestamps were recorded for the same event in both the smartphone and AVN (see **Table 5**). This indicates that one of timestamps in the smartphone or AVN was tampered. However, the Bluetooth log analysis alone could not determine which device’s timestamp had been manipulated. Therefore, to identify the source of the manipulation, further analysis of other logs, such as the system logs, is required.

TABLE 5. RESULT OF BLUETOOTH LOG ANALYSIS

Smartphone		
Timestamp	*Event Type*	*Log Message*
2024-09-05 15:41:59	Calling	Sent +CLCC: 1,1,0,0,0,”01049232198”,129
2024-09-05 15:42:14	Music	Sent Vendor dependent: Stable – GetElementAttributes – Title: “A Collection of 2000s hit”
Vehicle’s AVN		
Timestamp	*Event Type*	*Log Message*
08-28 13:41:06	Music	MobisAvrcpControllerService [BTAD] title: A Collection of 2000s hit, artist:, album:, playingTime:12345000

B. System Log

The results of the system log analysis from both the smartphone and the AVN system are shown in **Table 6**. The smartphone’s system logs contained the message from **'time_detector'** indicating **'automatic time enabled'** confirming that the smartphone was using automatic time settings. Thus, it was verified that the smartphone’s timestamps had not been manipulated.

TABLE 6. RESULT OF SYSTEM LOG ANALYSIS

Smartphone		
File Name	*Timestamp*	*Log Message*
dumpstate-2024-09-05-15-43-02.txt	09-05 15:40:19	automatic time enabled
Vehicle’s AVN		
File Name	*Timestamp*	*Log Message*
telematics.txt	08-28 13:40:00	Action : android.intent.action.TIME_SET
SET_USER_TIME@1724820000044	-	millis: 1724820000000 offset: [new] -698480811 [isUsetTimeSet] true

On the other hand, the AVN system’s logs included the **'android.intent.action.TIME_SET'** message in the

'telematics.txt' file, as well as the messages such as **'millis: 1724820000000'**, **'offset: [new] -698480811'** , and **'[isUsetTimeSet] true'** in the **'SET_USER_TIME@1724820000044.txt'** file. The message **'[isUsetTimeSet] true'** showed that the AVN's timestamp had been altered. After converting the values of the system log messages, we were able to determine the correct timestamp before the timestamp manipulation.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed a technique for detecting timestamp manipulation in an Android-based AVN system connected to an Android smartphone via Bluetooth. The proposed method involves collecting and analyzing log data from both devices. By analyzing the Bluetooth logs extracted from the two devices, we identified discrepancies in the timestamps for the same event, indicating potential problems associated with the timestamps. Additionally, by analyzing the system logs, we were able to identify log messages related to timestamp manipulation, allowing us to pinpoint which device's timestamp was tampered with.

The proposed technique can be applied to digital forensics to detect attempts to obstruct forensic investigations through timestamp manipulation. However, a limitation is that the Bluetooth HCI Snoop Log feature on Android phones is disabled by default, preventing the extraction of Bluetooth logs unless this feature is manually enabled. To overcome this limitation, future research will focus on developing a method to extract Bluetooth logs even when the Bluetooth HCI Snoop Log feature is disabled. This study was limited to a smartphone running Android 14 and an AVN system on Android 4.4.2. To improve the generalizability of the proposed technique, future work will test it across various Android versions and a broader range of target devices. Additionally, future research will investigate more extensive timestamp manipulation scenarios to enhance the technique's robustness and applicability.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (no. 2021R1A2C2012574), and by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by Korea government (MSIT) (No.1711170476, Development of Collection and Integrated Analysis Methods of Automotive Inter/Intra System Artifacts through Construction of Event-based experimental system).

REFERENCES

[1] André Årnes, "Digital Forensics: An Academic Introduction." John Wiley & Sons Inc, Hoboken, NJ, 2018.

[2] Gyu-Sang Cho, "Digital Forensic Analysis of Times tamp Change Tools: An Anti-Forensics Perspective." K orean Society of Computer Information Conference, 07 a, Pages.391-392, 2019.

[3] D. -i. Jang, G. -J. Ahn, H. Hwang, and K. Kim, "Understanding Anti-forensic Techniques with Timestamp Manipulation (Invited Paper)," 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI), Pittsburgh, PA, USA, pp. 609-614, 2016.

[4] Jewan Bang, Byeongyeong Yoo, and Sangjin Lee, "Analysis of changes in file time attributes with file manipulation", Digital Investigation, Volume 7, Issues 3–4, Pages 135-144, 2011.

[5] Automotive Electronics Magazine, "Smart car's infotainment service and smartphone connection technology,"https://www.autoelectronics.co.kr/article/articleView.asp?idx=933.

[6] R. Nusser and R. M. Pelz, "Bluetooth-based wireles s connectivity in an automotive environment," Vehicula r Technology Conference Fall 2000. IEEE VTS Fall VT C2000. 52nd Vehicular Technology Conference (Cat. N o. 00CH37152), Vol. 4, pp.1935-1942, 2000.

[7] H.I Kang, M.S Park, S.J Cho, and J.H Jung, "Using Logs of an Android-based Audio Video Navigation System for a Timeline Analysis in Vehicle Digital Forensics." Papers of the Korea Information Society 2023 Comprehensive Computer Science Conference, 1,259-1,261. 2023.

[8] J. Oh, S. Lee, and H. Hwang, "Forensic Detection of Timestamp Manipulation for Digital Forensic Investigation," in IEEE Access, vol. 12, pp. 72544-72565, 2024.

[9] M. Kaart and S. Laraghy, "Android forensics: Interpretation of timestamps," in Digital Investigation, Volume 11, Issue 3, Pages 234-247, 2014.

[10] H. Pieterse, M. S. Olivier, and R. P. van Heerden, "Playing hide-and-seek: Detecting the manipulation of Android Timestamps," *2015 Information Security for South Africa (ISSA)*, Johannesburg, South Africa, pp. 1-8, 2015.

[11] https://source.android.com/docs/core/tests/debug/read-bug-reports?hl=ko#event-log.

[12] Aya Fukami, Saugata Ghose, Yixin Luo, Yu Cai, and Onur Mutlu, "Improving the reliability of chip-off forensic analysis of NAND flash memory devices,Digital Investigation," Volume 20, Supplement, Pages S1-S11, 2017.

[13] V. Ndatinya, "Network forensics analysis using Wireshark", International Journal of Security and Networks(USN), Vol.10, No. 2, 2015.