electronics

MDPI

Article

Smart Home Forensics—Data Analysis of IoT Devices

Soram Kim [1], Myungseo Park [1], Sehoon Lee [1] and Jongsung Kim [1,2,*]

[1] Department of Financial Information Security, Kookmin University, 77 Jeongneung-Ro, Seongbuk-Gu, Seoul 02707, Korea; kimsr2040@kookmin.ac.kr (S.K.); pms91@kookmin.ac.kr (M.P.); dreamtree304@kookmin.ac.kr (S.L.)

[2] Department of Information Security, Cryptology and Mathematics, Kookmin University, 77 Jeongneung-Ro, Seongbuk-Gu, Seoul 02707, Korea

* Correspondence: jskim@kookmin.ac.kr

Received: 10 June 2020; Accepted: 24 July 2020; Published: 28 July 2020

Abstract: A smart home is a residence that provides a variety of automation services based on Internet of Things (IoT) devices equipped with sensors, cameras, and lights. These devices can be remotely controlled through controllers such as smartphones and smart speakers. In a smart home, IoT devices collect and process data related to motion, temperature, lighting control, and other factors and store more diverse and complex user data. This data can be useful in forensic investigations but it is a challenge to extract meaningful data from various smart home devices because they have different data storage methods. Therefore, data collection from different smart home devices and identification and analysis of data that can be used in digital forensics is crucial. This study focuses on how to acquire, classify, and analyze smart home data from Google Nest Hub, Samsung SmartThings, and Kasa cam for forensic purposes. We thus analyzed the smart home data collected using companion apps, Web interfaces, and APIs to identify meaningful data available for the investigation. Moreover, the paper discusses various types of smart home data and their usage as core evidence in some forensic scenarios.

Keywords: digital forensics; smart homes; data analysis; internet of things

1. Introduction

A smart home provides a variety of automation services based on Internet of Things (IoT) devices connected to a central hub or a gateway, which can be remotely controlled by controllers such as smartphones, tablets, and smart speakers. Each smart home device provides an independent service, while its companion apps, typically installed on smartphones, can be used to operate and monitor the device. In addition to smartphones, smart speakers such as Google Nest Hub and Alexa can be used as controllers of smart home devices connected to a smart home via voice commands.

Data from devices such as smart speakers, fitness wearables, pacemakers, and biometric devices are often used in courts as evidence during trials. For example, a file recorded by a a smart speaker played a crucial role in proving the innocence of a murder suspect in 2015 [1]. In the same year, records from a fitness tracker were used to establish that the statement of a suspect was false [2]. In 2017, pacemaker data were used as evidence to prove an insurance fraud [3].

IoT devices exist in various forms in-service platforms such as smart homes, smart cities, smart farms, and AMR systems. Devices belonging to each platform process data in a unique storage and management method, and have a relationship between data due to a connection between devices. Therefore, in order to utilize these data as evidence, it is necessary to analyze the relationship of the data of the connected devices as well as each device.

In this paper, we analyze data collected from companion apps, web interfaces, and APIs for smart home, one of the IoT service platforms, and propose digital forensic scenarios to utilize them.

We identify the sources and the format of smart home data and classified useful data. In addition, we perform correlational analysis between data to obtain accurate and diverse data. This study focuses on using data in forensic investigations.

1.1. Our Framework for Smart Home Forensics

Figure 1 illustrates the workflow of our framework for smart home forensics. The workflow starts with identifying the detailed functions of each smart home device and conducting experiments to accumulate data. The functional profiling of each smart home device is important for conducting efficient experiments and generating investigation artifacts. The second step involves collecting accumulated data. Due to the diversity of devices in a smart home, several data extraction methods, one for each device, should be considered. The third step involves analyzing smart home data. In particular, correlational analysis is performed in this step to identify the relationships between acquired data and user behavior. The final step involves employing analyzed data in forensic investigations. In this step, the meaning of each data source is established and data to be correlated are identified.

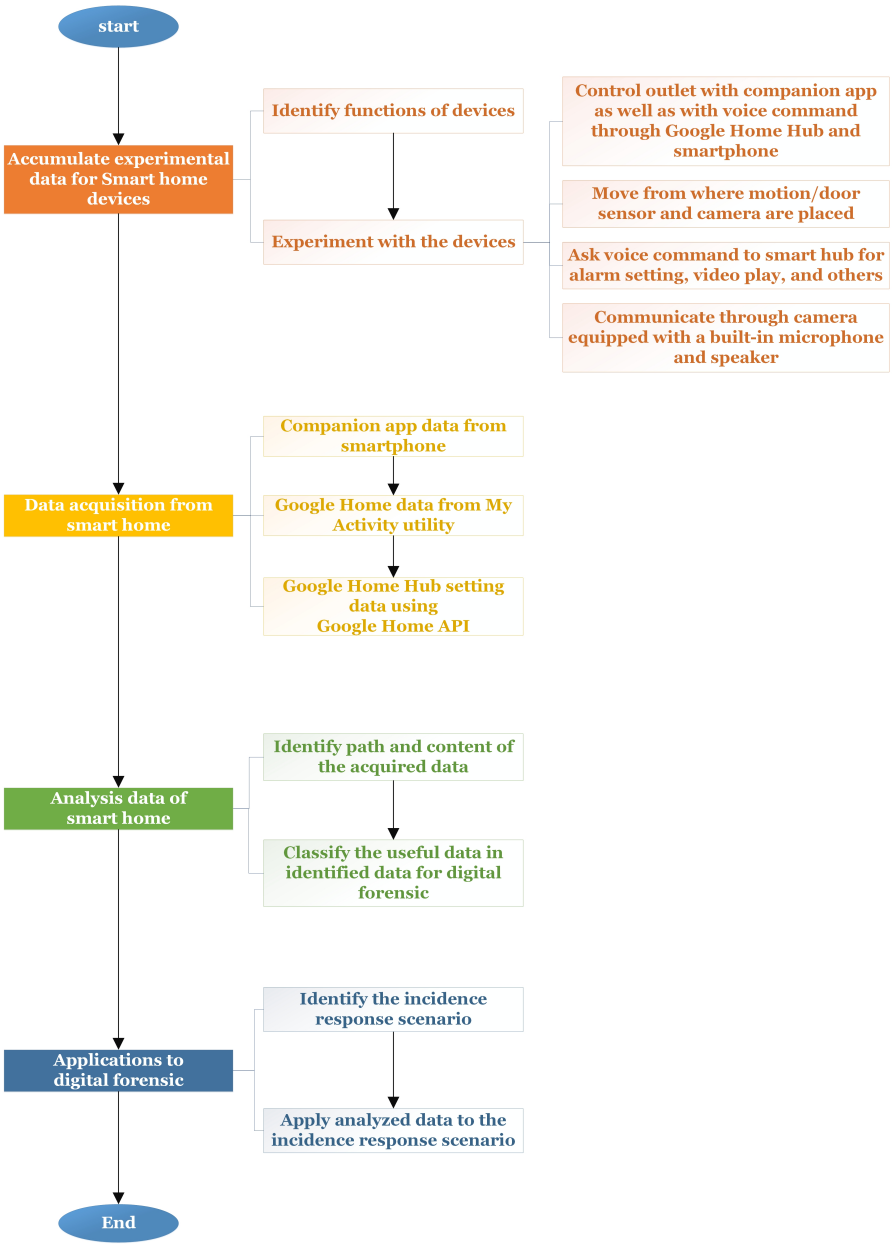

Figure 1. Workflow of the smart home forensic investigation.

1.2. Contributions

This study demonstrates that a variety of smart home data can be used in forensic investigations and suggests scenarios of applying digital forensics to smart home devices. In particular, the contributions of this study are the following.

(a) Identifying the data that can be used as forensic evidence and establishing the sources of these data. The data is identified and classified based on the functions of each smart home device;
(b) Describing the ways of applying analyzed data to some specific forensic scenarios. Examples of data sources in theses scenarios include smart home devices, movements, voice commands, and the call history of incident responses;
(c) Performing correlational analysis for obtaining user information, including event-specific time and user behavior information, and employing the relationships between data elements in digital forensics.

Table 1 lists the data considered in this study. 'Item' refers to an app or a device list while 'Source' refers to the origin of data. 'Data type' refers to the format of the acquired data; one potential scenario where this information can be used is discussed in Section 6.

Table 1. Summary of the data considered in the study.

Item	Source	Data Type	Forensic Information
Google Nest Hub	app, Web, API	xml, protobuf, json, mp3	voice command, device list
Google Duo	app	DB	call
TP-Link	app	image, xml, DB	movement
Samsung SmartThings Classic	app	Archive, log	movement
Samsung SmartThings	app	DB	movement, device list

2. Related Work

Analyzing data generated by smart homes for forensic purposes involves data extraction, data analysis, and an app with forensic tools for a variety of devices such as smart TVs, and smart speakers. The development of such forensic tools, including methods for include acquiring root privileges, pre-imaging, testing, post-imaging, and comparing binary data from smart TV forensics to trace user behavior is described in Reference [4]. A method for extracting data from a smart TV using embedded MultiMediaCard (eMMC) chips and companion apps is proposed in Reference [5]. The extraction and analysis of data from the fitness bands Xiaomi Mi Band2 and Fitbit Alta HR are described for monitoring user activity and sleep, together with schemes for recovering deleted data, are described in Reference [6]. Li et al. proposed an IoT-based forensic model and verified it using the Amazon Echo [7]. Seila et al. acquired and analyzed data of a Samsung Gear S3 Frontier smart watch [8].

Building on these studies, research has been conducted on the collection and analysis of data from controllers controlling one or more devices, rather than merely analyzing data from a single device. Gokila et al. examined the Nest artifacts produced by an iPhone and developed a forensic tool called the Forensic Evidence Acquisition and Analysis System [9]. Guidance on the forensic data acquisition and analysis of artifacts from Securifi Almond+, another smart home hub environment, was reported in Reference [10]. Amazon Echo, a smart speaker widely used as a smart device controller, was forensically analyzed by a proof-of-concept tool called Cloud-based IoT Forensic Toolkit [11]. Douglas et al. suggested that the evidentiary value of data from Amazon Echo is considerable [12].

Fensel et al. proposed an IoT semantic platform called OpenFrigde which incorporated the IoT, Web, and Semantic Web technologies, and enabled users to collect, process, link, and view various IoT data [13].

Many digital forensic frameworks have been proposed to support investigation and provide guidelines [14–17]. Other survey papers [18–24] have focused on the requirements, approaches, and challenges of IoT forensics.

This study considers new devices not previously studied to utilize various IoT devices in forensic investigations. The existing studies that are focused on single device analysis are difficult to utilize in a smart home environment connected with various IoT devices. In this study, we perform correlation analysis by extracting data of various IoT devices in the smart home, and present scenarios that can utilize their data in digital forensic investigations.

3. Collecting Experimental Data from Smart Home Devices

Figure 2 illustrates the smart home environment purposely built for this study to enable data collection. The environment included a Google Nest Hub, a Kasa Cam, a SmartThings Multi-purpose Sensor, a SmartThings Motion Sensor, and a SmartThings Outlet for smart homes [25–27].

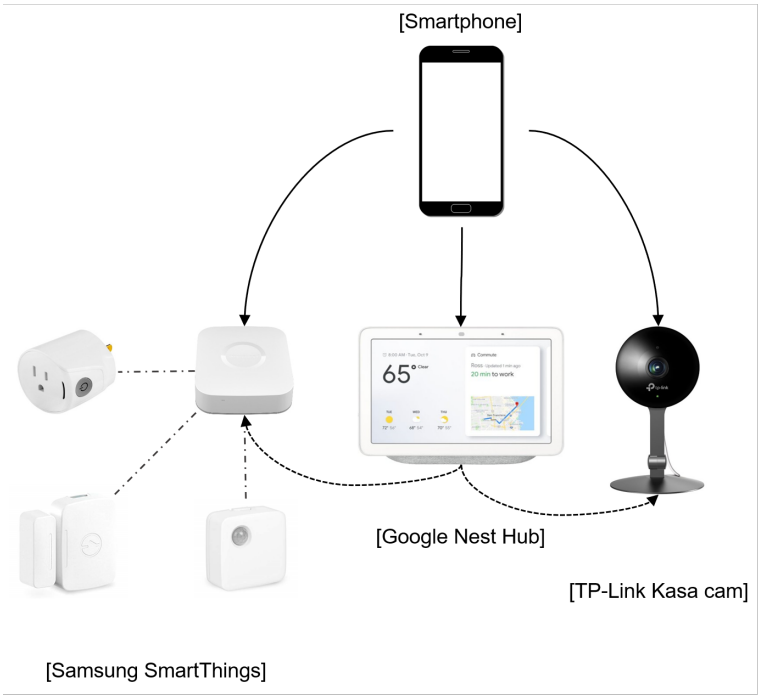

Figure 2. Target smart home devices.

The following steps were completed to collect data.

(a) Device activation;
(b) Companion app download & device enrollment;
(c) Identifying functions of devices;
(d) Experiment with the devices.

The SmartThing Hub connected to the low-end smart home devices, serves as a base station for the data exchange between the home devices. Each smart home kit employed in the experiment can be used alone or in conjunction with a supported smart speaker. In this study, the Google Nest Hub was employed as a smart speaker given its popularity. The installed smart home devices can be controlled by a smart speaker using voice commands or an app on a smartphone. Table 2 lists the devices employed in the experiment.

Table 3 lists the features and functions of each device used in the experiments. The Google Nest Hub can control numerous elements of the home, from an outlet to a camera. The hub is equipped with a display, making it possible to see and control connected devices in a single window, with no need to switch between apps. The display can also show videos from the smart camera, Kasa cam. Calls between a smartphone and the Google Nest Hub are possible through the video calling Google Duo app. The Kasa cam can be watched live; alternatively, the feed can be recorded on a smartphone

using the Kasa Smart app. When motion and sounds are detected, the camera sends a notification to a smartphone and saves the video. The cam has a built-in microphone and speaker enabling two-way communication through the camera and its companion app. The SmartThings Outlet can control whether power is on or off. The SmartThings Multipurpose Sensor can detect open and closed status, as well as vibration. The multipurpose sensor and motion sensor can both detect temperature.

Table 2. Test devices employed in the experimental environment.

Device	Model
(1) Google Nest Hub	H1A 8C012AA19DIW
(2) Kasa Cam	KC 120
(3) SmartThings Outlet	STS-OUT-US-2
(4) SmartThings Multipurpose Sensor	STS-MLT-250
(5) SmartThings Motion Sensor	STS-IRM-250
(6) Samsung Galaxy Note 9	SM-N960N

Table 3. Functions of smart home devices.

Device	Functions of Device
Google Home	- Control smart home devices via voice commands, the display, or companion app - Provide visual services such as music and video playback, photo display,reminder, and search - Make calls between a smartphone and the Google Nest Hub using the duo app
Kasa Cam	- Watch live or record video - Detect motion and sounds - Two-way communication through the camera and companion app
SmartThings Outlet	- Power on/off - Measure power consumption
SmartThings Multipurpose Sensor	- Detect open or close status - Detect temperature - Detect vibration
SmartThings Motion Sensor	- Detect motion - Detect temperature

4. Data Acquisition from Smart Home Devices

In this study, companion app data for Android smartphones were collected from smart home devices. Google Nest Hub data were obtained from its application programming interface (API) and Web pages returned by the Google.

4.1. Companion Apps

The manufacturers of smart home devices provide companion apps that are synchronized with the devices to enable the device control, management, and configuration. Due to the limited storage capacity of smart home devices, information about the operation of these devices is generally stored on a smartphone under the companion app package name folder in the data partition. Table 4 lists the package names and versions of the apps used in this study. The SmartThings and SmartThings Classic apps support Samsung smart home devices. The SmartThings Classic app is an old version, and the SmartThings app is a new version.

Table 4. Companion apps for smart home devices.

App Name	Package Name	Version
Google Home	com.google.android.apps.chromecast.app	2.16.1.10
SmartThings	com.samsung.android.oneconnect	1.7.41-25
SmartThings Classic	com.smartthings.android	2.18.1
Kasa Smart	com.tplink.kasa_android	2.18.0.900
Google Duo	com.google.android.apps.tachyon	69.0.286207624.DR69_RC12

4.2. Google Web Interface

The Google Web interface utility called My Activity was used to generate data, including the service usage history for applications such as YouTube, Google Maps, the Chrome browser, and a variety of other apps. The Google Nest Hub voice commands were utilized to obtain data about the use of Google services but accessing this utility, a Google account is required.

The path we used to obtain the Google product data in My Activity was 'Google Account > Data & personalization > Download your data.' The following steps were performed to acquire data:

(a) Select data

Product data is stored in 'Home App' and 'My Activity'. Home App data can be exported only in the JSON format, whereas My Activity data can be exported in either the HTML or JSON format.

(b) Customize archive format

A link for downloading data can be sent by email, or the data can be added to a drive, Dropbox, One Drive, or Box. The file type is either .zip or .tgz. Large archives can be split into multiple files.

(c) Output generated file

The output file name format is 'YYYYMMDDThhmmssZ-number.' The field 'number' indicates the order if acquired data is split across multiple files. The timestamp of the file name is set to UTC+0.

4.3. Private Google Home API

The Google Home app provides the ability for local APIs to communicate with devices. While the majority for the APIs are not publicly available, we learned how to use the APIs by referring to private documents that are available through communities such as GitHub. Data from Google Home was obtained using the HTTP method of the local APIs of Google Home as described in Reference [28]. The GET method was used to obtain data from Google Home instead of the POST method; the latter is mainly used for changing settings. The GET method performs a request through a browser. When the request is concatenated to the base URL and sent to the server, the response data can be received in the JSON format. Since Google Home uses the server port 8008, a request can be sent in the form of http://(google-home-ip):8008/(request). Here the Google Home IP can be identified via the Google Home app (Device Settings-Info) or the Google Nest Hub (System Settings).

5. Analyzing Data Generated by Smart Home Devices

This section presents the analysis of the data obtained from companion apps, Web interfaces, and an API. In particular, the stored path and content from the collected data was identified and the useful data was classified.

5.1. Companion Apps

Each companion app can register a respective smart home device and then manage activity records generated whenever the device detects an action. Table 5 lists the file paths of data stored by smart home apps.

Table 5. Path and data of smart home devices.

Device	
Path	**Content**
Google Nest Hub (/data/data/com.google.android.apps.chromecast.app)	
/shared prefs/com.google.android.apps.chromecast .app preferences no backup.xml	Location, Account, Wi-Fi SSID
/files/home_graph_value of encoded user account using base64.proto	Homegroup, Registered Devices
Google Duo (/data/data/com.google.android.apps.tachyon)	
/databases/tachyon.db	Call history(between smartphone and Google Nest Hub)
TP-Link Cam (/data/data/com.tplink.kasa android)	
/cache/image manager disk cache/32bytes_hex _value.0	Thumbnail
/cache/image manager disk cache/Kasa.xml	DeviceAddress, DeviceAlias, DeviceCategory, DeviceId, DeviceModel, DeviceType, FirmwareId, HardwareId, HardwareVersion, IPAddress
/databases/iot.db	Accounts, Registered Devices, Locations
Samsung SmartThings Classic (/data/com.smartthings.android)	
/cache/http/16bytes_hex_value.0	Information about file 16bytes_hex_value.1
/cache/http/16bytes_hex_value.1	Detected sensor motion
Samsung SmartThings (/data/com.samsung.android.oneconnect)	
/databases/ActivityLogDb.db	Activity log
/databases/CloudDb.db	Registered devices, group, location
/databases/Favorite.db	Favorite devices
/databases/InternalSettings.db	User information
/databases/NotificationDb.db, /databases/NotificationUIDb.db	Notification information

5.1.1. Google Nest Hub

Google Nest Hub stores data about location, accounts, Wi-Fi SSID, homegroup, and registered devices on a smartphone; information such as voice commands is not recorded as part of the app data. The home_graph_Base64Encoding (Google account).proto file, which contains the homegroup and registered device information, uses an encoded protocol buffer to store the data; hence, decoding is required to confirm the content. Decoding can be performed using an online decoding tool [29].

The call history between a smartphone and a Google Nest Hub is stored as duo app data. In tachyon.db, the [activity_history] stores the call history. other_id column stores the phone number of the opposite party and self_id column stores the user phone number. Google Nest Hub is set the same phone number and Google account. If other_id and self_id are the same, it implies that the call occurs between the smartphone and the Google Nest Hub. The value 1 of in call_state indicates that a call failed, whereas the value of 2 means that the call succeeded. The call duration can be obtained from basic call app data.

5.1.2. Kasa Cam

The TP-Link camera stores videos recorded by users or automatically when detecting motion or sounds. Given a limit of 1 GB memory, the video history is stored for up to two days less than four hours of 1080p video by default. However, video data is stored only if users download them. Otherwise, only the thumbnail file remains. The video title format is 'KC_Camera name_ddmmyyyy_msmsmsms.mp4', and the downloaded video files are stored in the gallery. Thumbnails represent either the video recorded when motion is detected or images were taken when the live video is viewed through the smartphone app. The former generates a .PNG file, while the latter generates a .JPG file. However, since the extension of the thumbnail recorded in the app data is

.0, it is necessary to change the extension after checking the signature to be able to view the image. The camera is equipped with a built-in microphone and speaker, allowing two-way communication with the camera and smartphone companion app.

5.1.3. Samsung SmartThings

The SmartThings door and motion sensor can recognize the open and close states of door, motion, and temperature. When the temperature is changed, the sensor sends the new data to a smartphone, where it is then stored. The outlet also records data about the power consumption and state of power being on or off.

SmartThings Classic App

The http directory contains two files, 16bytes_hex_value.0 and 16bytes_hex_value.1. The 16bytes_hex_value.0 file stores information about the 16bytes_hex_value.1 file such as its content type, encoding status, and timestamp. If the content-encoding field does not exist, the file is not encoded. If the type of 16bytes_hex_value.1 file is Gzip, a JSON file can be acquired by unzipping the file. The JSON file contains the sensor name, date, Unix time, and sensor event. The date is expressed as UTC + 0, so it might need to be converted to the desired time zone.

SmartThings App

Unlike the Samsung SmartThings Classic app, all data generated by the Samsung SmartThings app are stored in databases. In ActivityLogDb.db, the [activityLog] and [deviceTable] contain the logs of sensor activities. In CloudDb.db, the [devices], [groups], and [locations] columns of the database include the registered devices, group information, and location, respectively. In Favorite.db, [Favorite] contains the favorite devices set by the user. In InternalSettings.db, [isettings] contains user information such as user name and ID. In NotificationDb.db and NotificationUIDb.db, [messages] and [messagesUI] contain notification information generated when the app alerts the smartphone.

5.2. Google Web Interface

My Activity shows the user's activity history in chronological order. Item Details contain voice commands, voice records, and answers by the Google Nest Hub, time, device type, and location. Every command to the Google Nest Hub is recorded as a voice command, but voice records exist only for the voice commands. Location, defined by the longitude and latitude, shows the corresponding Google Map position.

My Activity and Google Home data acquired from the Web is stored in MyActivity.json or MyActivity.html files according to the option chosen. 'Assistant' and 'Voice and Audio' directories contain .mp3 files of voice commands; the file name format is YYYY-MM-DD_hh_mm_fff_UTC.mp3.

The HomeApp.json file contains all Google Nest Hub app data such as owner emails and homes, rooms and location, and the details of mapping devices to rooms.

The My Activity > Voice and Audio > MyActivity.json file contains objects comprising a header, a title, a titleUrl, time, audioFiles, products, and details. The meaning of each value is as follows:

- Header: target of the command;
- Title: user command;
- TitleUrl: URL of the answer to the search query;
- Time: time of the occurrence;
- AudioFiles: name of the saved audio file;
- Products: categorization;
- Details: action command inside the device.

The My Activity > Assistant > MyActivity.json file contains a header, a title, a titleUrl, a subtitle, time, products, and locations (name, url). The meaning of each value that is not defined above is as follows:

- Subtitle: answer to the user command
  - Name: answer to the user command;
  - Url: URL of the answer to the search query.
- Locations: device location information
  - Name: action inside the device;
  - Url: Google Map URL of the location.

Voice and Audio directory store not only the voice data processed through the assistant but also the data used for voice learning, such as voice match. The Assistant directory is therefore a subset of the Voice and Audio directory, but the MyACtivity.json file of the Assistant is more detailed than the Voice and Audio.

5.3. Private Google Home APIs

Google Nest Hub data can be obtained throgh private APIs and the Web in the JSON format. Table 6 list the APIs selected as useful for digital forensics among all available private Google Home APIs. The output content type is JSON, so we can easily distinguish the data.

Table 6. List of private Google Home APIs.

Category	Type	Path	Content
Device Information	Ereka Info	/setup/eureka_info	Build information
	Timezones	/setup/supported_timezones	Timezone
Bluetooth	Status	/setup/bluetooth/status	Connecting Devices & Connected devices
	Get Paired Devices	/setup/bluetooth/get_bonded	All of paired devices
	Get Scan Results	/setup/bluetooth/scan_results	List of all nearby bluetooth devices
Wi-Fi	Get Saved Networks	/setup/configured_networks	List of all saved Wi-Fi networks

Both the reboot and factory reset functions, which can act as anti-forensic approaches, are available using the private Google Home APIs. Rebooting can cause the data remaining in the memory to disappear, while factory reset deletes all information, including user accounts and setting information, and others. These two functions can be exploited by an attacker who is on the same network as the Google Nest Hub and knows its IP address. The reboot and factory reset functions are available with tools such as curl using the POST method.

6. Exploiting Smart Home Data in Digital Forensics

This section outlines possible scenarios of utilizing smart home data described in the previous sections in digital forensics. The focus is on data generated by the motion/door sensor and AI speaker for intrusion detection. Correlation analysis is demonstrated to provide insights into the meaning of the data.

6.1. Information from Smart Home Devices

Due to the miniaturization and diversity of smart home devices, it is difficult to identify the ones that can be useful for digital forensics. The configuration information of a smart home can be exploited as the basis for identifying smart home devices. The .proto file (Google Home app) and the HomeApp.json file (Google My Activity utility) contain information about the smart home that is organized around the Google Nest Hub. The files can provide such details as the name of the

smart home, information about each room, and a list of devices installed in each room. In Samsung SmartThings, the device information is included in the 'devices' table of cloudDb.db. Based on these details, the investigator can decide which devices and data to target.

6.2. Information about Movement

Data about movement can help to determine the time of the invasion in the case of an intrusion event. For example, such data can be used to determine the actual time of the raid and establish if the house owner made a false statement.

The multipurpose SmartThings sensors, which can be used as door or window sensors, detect the open and close states of the objects they attached to. The data collected by each such sensor is recorded in ActivityLogDb.db (Samsung SmartThings app) and /cache/http/16bytes_hex_value.1 (Samsung SmartThings Classic app), which includes the following:

- Sensor acceleration: active/inactive;
- Sensor status: open/closed.

These data can provide information about whether or not a person was in a specific place at a particular time and the duration of the stay. Motion data can be obtained from two devices, the motion sensor of the Samsung SmartThings and the TP-Link camera. In the Samsung SmartThings, motion sensor data such as the time of motion start and finish and temperature is stored in the same file as data from the multipurpose sensor. The TP-Link camera stores thumbnails of videos recorded when motion occurs in /cache/image manager disk cache/32bytes_hex_value.0. These thumbnails can provide information about the appearance or shape of a suspect in a housebreaking incident, for example.

By combining motion and door sensor information, it is possible to determine whether a person has entered or left the house. The open and close event of the door sensor occurred before or after motion detection indicates that someone has entered or left the home, respectively.

6.3. Information about Voice Commands

Voice information can be very useful for identification. Voice files are generated when voice commands are used to control devices or retrieve information. These data provide information about the activity history and voices of users. For example, suppose a person identified as a suspect in crime used a voice command from a Google Nest Hub to get directions to the crime scene. Investigators could identify the voice through the recorded file. Voice command data can also be used for purposes such as verifying statements or understanding the situation at the site.

6.4. Information about Calls

The user call logs, which include the call duration and information about the other party, can be used as very important evidence. If a user has a call log with the Google Nest Hub, investigators can guess that someone was home at that time. To arrange a call between the Google Nest Hub and a smartphone, the user can use the "Call my Home devices" function within the Google Duo app. In this case, the phone numbers for other_id and self_id would be the same in the call log of the Duo app. If the use tries to call his/her mobile number using the default system call app, the call will be directed to the voicemail. This means that the call log in the default system call app does not allow to identify whether the user has tried to call the Google Nest Hub or check the voicemail. In this case, investigators need to compare the default system call app and Google Duo app data to identify the purpose of the call. If the call record exists in the default system call app but not in the Google Duo app, this means that the call has been connected to the voicemail. If not, this indicates that the user has tried to call the Google Nest Hub.

Figure 3 visualizes the sources for obtaining different types of information.

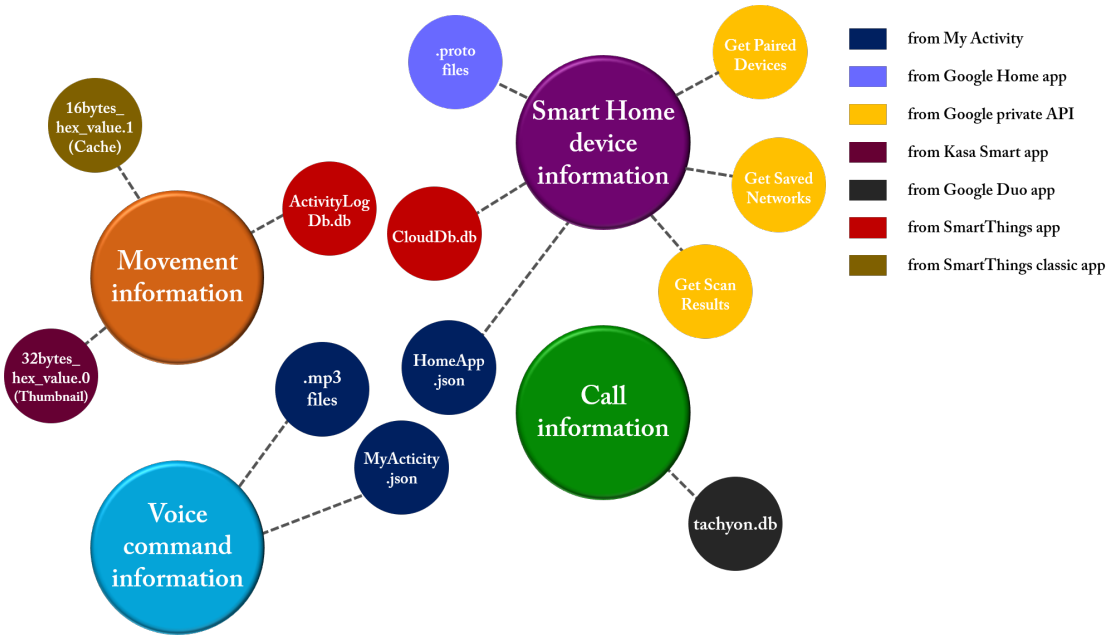

Figure 3. Data classification per information.

The Google Home, SmartThings, SmartThings Classic, and Google Duo apps (but not the Kasa Smart app) do not allow users to delete the user data in the app unless deleting the app or the account; hence users cannot arbitrarily manipulate the data. Therefore, the data sources identified in this study are expected to be useful for providing convincing evidence in courts.

7. Conclusions and Future Development

This study looked at the data that can be acquired from smartphones, the Google Web interface, and private Google Home APIs, and identified the ones that can be useful for digital forensic analysis. Companion apps installed on smartphones and paired with smart home devices can provide information on the location and model of each device. The activity history recorded by each device can be useful for forensic investigations. For example, the statement of a suspect can be verified through motion, temperature, command, and voice data, which are also useful for inferring the time of an event occurrence. This study identified the sources of various types of data generated by a set number of smart home devices that can be cross-checked to establish the actual sequence of events. Future work should focus on analyzing a wider range of smart home devices to cover many different scenarios during a forensic investigation since the type and amount of data generated by smart home devices varies across device manufacturers and models. Moreover, the investigation of the physical memory forensic of smart home devices should be included.

Author Contributions: Conceptualization, Supervision, S.K. and M.P.; investigation, S.K. and S.L.; writing—original draft preparation, S.K.; writing—review and editing, M.P., J.K.; project administration, J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Korea government (MSIT, Ministry of Science and ICT).

Acknowledgments: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2019R1F1A1060634).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Arkansas Judge Drops Murder Charge in Amazon Echo Case. Available online: https://edition.cnn.com/2017/11/30/us/amazon-echo-arkansas-murder-case-dismissed/index.html (accessed on 1 June 2020).
2. Cops Use Murdered Woman's Fitbit to Charge Her Husband. Available online: https://edition.cnn.com/2017/04/25/us/fitbit-womans-death-investigation-trnd/index.html (accessed on 1 June 2020).
3. Pacemaker Could Hold Key in Arson Case. Available online: https://edition.cnn.com/2017/02/08/us/pacemaker-arson---trnd/ (accessed on 1 June 2020).
4. Kang, H.-S.; Park, M.-S.; Kim, S.-J. Study on Smart TV Forensics. *J. Korea Inst. Inf. Secur. Cryptol.* **2014**, *24*, 851–860. [CrossRef]
5. Boztas, A.; Riethoven, A.R.J.; Roeloffs, M. Smart TV forensics: Digital traces on televisions. *Digit. Investig.* **2015**, *12*, 72–80. [CrossRef]
6. Kang, S.; Kim, S.; Kim, J. Forensic analysis for iot fitness trackers and its application. *Peer-to-Peer Netw. Appl.* **2020**, *13*, 564–573 [CrossRef]
7. Li, S.; Choo, K.-K.; Sun, Q.; Buchanan, W.J.; Cao, J. IoT forensics: Amazon Echo as a use case. *IEEE Internet Things J.* **2019**, *6*, 6487–6497. [CrossRef]
8. Becirovic, S.; Mrdovic, S. Manual IoT Forensics of a Samsung Gear S3 Frontier Smartwatch. In Proceedings of the 2019 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 19–21 September 2019; pp. 1–5.
9. Dorai, G.; Houshmand, S.; Baggili, I. I know what you did last summer: Your smart home Internet of Things and your iPhone forensically ratting you out. In Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg, Germany, 27–30 August 2018.
10. Awasthi, A.; Read, H.O.; Xynos, K.; Sutherland, I. Welcome pwn: Almond smart home hub forensics. *Digit. Investig.* **2018**, *26*, 38–46. [CrossRef]
11. Chung, H.; Park, J.; Lee, S. Digital forensic approaches for Amazon Alexa ecosystem. *Digit. Investig.* **2017**, *22*, 15–25. [CrossRef]
12. Orr, D.A.; Sanchez, L. Alexa, did you get that? Determining the evidentiary value of data stored by the Amazon® Echo. *Digit. Investig.* **2018**, *24*, 72–78. [CrossRef]
13. Fensel, A.; Tomic, D.K.; Koller, A. Contributing to appliances' energy efficiency with Internet of Things, smart data and user engagement. *Future Gener. Comput. Syst.* **2017**, *76*, 329–338. [CrossRef]
14. Goudbeek, A.; Choo, K.K.; Le-Khac, N.A. A Forensic Investigation Framework for Smart Home Environment. In Proceedings of the 2018 17th IEEE International Conference on Trust, Security And Privacy in Computing And Communications/12th IEEE International Conference on Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, 1–3 August 2018; pp. 1446–1451.
15. Alharbi, R.; Allen, W.H. Collection and Analysis of Digital Forensic Data from Devices in the Internet of Things. In Proceedings of the 2019 SoutheastCon, Huntsville, AL, USA, 11–14 April 2019.
16. Islam, M.J.; Mahin, M.; Khatun, A.; Debnath, B.C.; Kabir, S. Digital Forensic Investigation Framework for Internet of Things (IoT): A Comprehensive Approach. In Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 3–5 May 2019; pp. 1–6.
17. Philomin, S.; Singh, A.; Ikuesan, A.; Venter, H. Digital Forensic Readiness Framework for Smart Homes. In Proceedings of the International Conference on Cyber Warfare and Security, Islamabad, Pakistan, 29 September–1 October 2020; pp. 627–638.
18. MacDermott, A.; Baker, T.; Shi, Q. Iot forensics: Challenges for the ioa era. In Proceedings of the 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 26–28 February 2018; pp. 1–5.
19. Zulkipli, N.H.; Alenezi, A.; Wills, G.B. IoT Forensic: Bridging the Challenges in Digital Forensic and the Internet of Things. In Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security (IoTBDS 2017), Porto, Portugal, 24–26 April 2017; pp. 315–324.
20. Stoyanova, M.; Nikoloudakis, Y.; Panagiotakis, S.; Pallis, E.; Markakis, E.K. A Survey on the Internet of Things (IoT) Forensics: Challenges, Approaches and Open Issues. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1191–1221. [CrossRef]

21. Servida, F.; Casey, E. IoT forensic challenges and opportunities for digital traces. *Digit. Investig.* **2019**, *28*, 22–29. [CrossRef]
22. Kruger, J.-L.; Venter, H. Requirements for IoT Forensics. In Proceedings of the 2019 Conference on Next Generation Computing Applications (NextComp), Mauritius, 19–21 September 2019; pp. 1–7.
23. Yaqoob, I.; Hashem, I.A.T.; Ahmed, A.; Kazmi, S.A.; Hong, C.S. Internet of things forensics: Recent advances, taxonomy, requirements, and open challenges. *Future Gener. Comput. Syst.* **2019**, *92*, 265–275. [CrossRef]
24. Yang, W.; Johnstone, M.N.; Sikos, L.F.; Wang, S. Security and Forensics in the Internet of Things: Research Advances and Challenges. In Proceedings of the 2020 Workshop on Emerging Technologies for Security in IoT (ETSecIoT), Sydney, Australia, 21–21 April 2020; pp. 12–17.
25. Google Nest Hub. Available online: https://support.google.com/googlenest/answer/9136909?hl=en/ (accessed on 1 June 2020).
26. Kasa Cam. Available online: https://www.kasasmart.com/us/products/security-cameras/kasa-cam-kc120/ (accessed on 1 June 2020).
27. SmartThings. Available online: https://www.samsung.com/us/smart-home/smartthings/ (accessed on 1 June 2020).
28. Online Protobuf Decoder. Available online: https://protogen.marcgravell.com/decode/ (accessed on 1 June 2020).
29. Private Documentation of the Local API Used by the Home App. Available online: https://rithvikvibhu.github.io/GHLocalApi/ (accessed on 1 June 2020).