

---

# **DFRWS IoT Forensic Challenge (2018-2019)**

# **Evidence Analysis Report**

---

Written by

**Team TapiocaPearlo**

<b>Gyuho Lee</b>	extr3.0@gmail.com
<b>Hyunpyo Choi</b>	glucose6126@gmail.com
<b>Namjun Kim</b>	bunseokbot@gmail.com
<b>Philgeun Jin</b>	andyjin777@gmail.com
<b>Sangho Park</b>	ddeok9@gmail.com
<b>Seho Kim</b>	s3hokim@gmail.com
<b>Seunghyeon Lee</b>	mpsniiper01@gmail.com
<b>Sohee Kim</b>	soheekim0927@gmail.com
<b>Suhyeon Jin</b>	jsh05042@gmail.com

# Contents

1	Introduction .....	1
1.1	Challenge Scenario & Questions .....	1
1.2	Overview of Acquired Data .....	3
2	Analysis Result Summary .....	4
2.1	Reconstruction of the Crime Scene .....	4
2.2	Answers to the Challenge Questions.....	9
3	Forensic Analysis Details .....	11
3.1	Device Level Analysis .....	11
3.1.1	Jessie Pinkman's Smartphone.....	11
3.1.2	iSmartAlarm Base Station .....	35
3.1.3	Arlo Base Station.....	37
3.1.4	Wink Hub.....	41
3.1.5	Amazon Echo .....	42
3.2	Network Level Analysis .....	45
3.2.1	Entities.....	45
3.2.2	Conversations .....	47
3.3	Correlations of Findings .....	49
3.3.1	Network Structure of the Drug Lab.....	49
3.3.2	Detailed Timeline.....	51
4	Developments .....	58
4.1	Initial Setup .....	58
4.2	Parser.....	60
4.3	Visualization .....	63
5	Appendix .....	68
5.1	List of Collected Evidence Files .....	68
5.2	Additional Explanation of the Forensic Analysis Details .....	71
5.2.1	Jessie Pinkman's Smartphone.....	71
5.2.2	iSmartAlarm Base Station .....	74
6	References.....	78

# 1 Introduction

This report summarizes the result and process of digital forensic analysis on evidences of DFRWS IoT forensic challenge (2018-2019) scenario.

The report is organized as follows: **Chapter 1** presents a general overview of the challenge scenario. **Chapter 2** reconstructs the crime scene based on information found through digital forensic analysis and deduces the answers to the challenge questions. **Chapter 3** describes the detailed analysis of evidences, and **Chapter 4** explains digital forensic analysis tools devised in the course of the challenge.

**Timezone of all timestamps appeared below are adjusted to UTC+2, unless otherwise noted.**

## 1.1 Challenge Scenario & Questions

### - Challenge Scenario

On 17 May 2018 at 10:40, the police were alerted that an illegal drug lab was invaded and unsuccessfully set on fire. The police respond promptly, and a forensic team is on scene at 10:45, including a digital forensic specialist.

The owner of the illegal drug lab, Jessie Pinkman, is nowhere to be found. Police interrogate two of Jessie Pinkman's known associates: D. Pandana and S. Varga. Pandana and Varga admit having access to the drug lab's Wi-Fi network but deny any involvement in the raid. They also say that Jessie Pinkman's had the IoT security systems installed because he feared attacks from a rival gang and that Jessie kept the alarm engaged in "Home" mode whenever he was inside the drug lab.

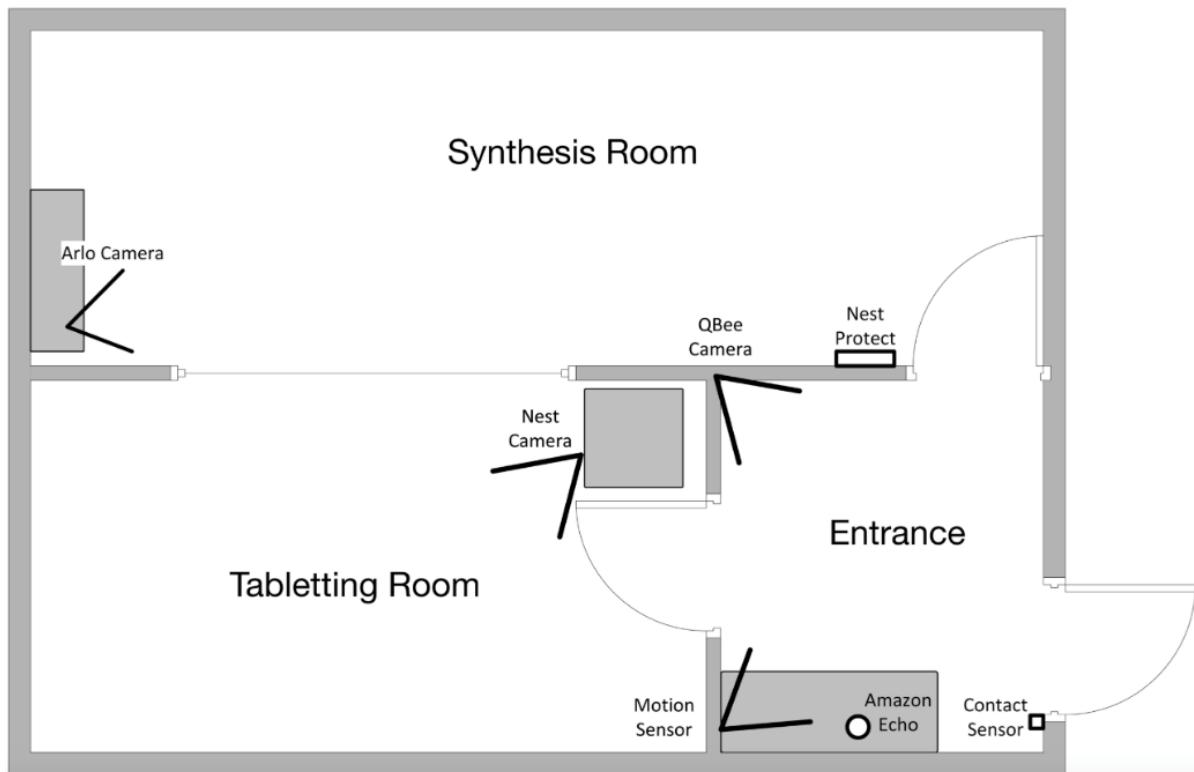
Within the drug lab the digital forensic specialist observes some IoT devices, including an alarm system (iSmartAlarm), three cameras (QBee Camera, Nest Camera and Arlo Pro) as well as a smoke detector (Nest Protect). An Amazon Echo and a Wink Hub are also present.

The digital forensic specialist preserves the diagnostic logs from the iSmartAlarm base station, and acquires a copy of the file system of the Wink Hub. He also collects the iSmartAlarm and Arlo base stations to perform an in-depth analysis at the forensic laboratory.

The digital forensic specialist also notices that the QBee Camera seems to be disabled, so he collects a sample of the network traffic.

Back at the forensic laboratory, the digital forensic specialist uses the bootloader to collect a memory image of the two base stations as well as an archive of some folder of interest of the Arlo base station.

Jessie Pinkman's Samsung Galaxy S6 Edge is found at the scene, likely dropped during the raid. The digital forensic specialist acquires a physical image of this Samsung device.



**Figure 1 – Diagram of the Drug Lab**

### - Challenge Questions

The Attorney General needs answers to the following questions:

- At what time was the illegal drug lab raided?
- Could any of the two friends of Jessie Pinkman have been involved in the raid?
  - If yes:
    - ◆ Which friend?
    - ◆ What is the confidence in such hypothesis?
- How was the QBee camera disabled?

## 1.2 Overview of Acquired Data

Responders acquired the following data (Time Zone: UTC+2):

**Table 1 – Description of acquired data**

Jessie Pinkman's Smartphone	
Description	Physical extraction
File Name	Samsung GSM_SM-G925F Galaxy S6 Edge.7z
SHA256	ae83b8ec1d4338f6c4e0a312e73d7b410904fab504f7510723362efe6186b757
iSmartAlarm Base Station	
Description	Diagnostic logs
File Name	ismartalarm/diagnostics/2018-05-17T10_54_28/server_stream
SHA256	8033ba6d37ad7f8ba22587ae560c04dba703962ed16ede8c36a55c9553913736
Description	Memory image: 0x0000'0000
File Name	dump/ismart_00.img
SHA256	ae83b8ec1d4338f6c4e0a312e73d7b410904fab504f7510723362efe6186b757
Description	Memory image: 0x8000'0000
File Name	dump/ismart_80.img
SHA256	ae83b8ec1d4338f6c4e0a312e73d7b410904fab504f7510723362efe6186b757
Arlo Base Station	
Description	Memory image
File Name	arlo/dfrws_arlo.img
SHA256	3b957a90a57e5e4485aa78d79c9a04270a2ae93f503165c2a0204de918d7ac70
Description	NVRAM settings
File Name	arlo/nvram.log
SHA256	f5d680d354a261576dc8601047899b5173dbbad374a868a20b97fdb963dca798
Description	NAND: TAR archive of the folder /tmp/media/nand
File Name	arlo/arlo_nand.tar.gz
SHA256	857455859086cd6face6115e72cb1c63d2bef11db92beec52d1f70618c5e421
Wink Hub	
Description	File system TAR archive
File Name	wink/wink.tar.gz
SHA256	083e7428dc1d0ca335bbcf11c6263720ab8145ffc637954a7733afc7b23e8c6
Amazon Echo	
Description	Extraction of cloud data obtained via CIIFT
File Name	echo/(2018-07-01_13.17.01)_CIIFT_RESULT.zip
SHA256	7ee2d77a3297bb7ea4030444be6e0e150a272b3302d4f68453e8cfa11ef3241f
Network Capture	
File Name	network/dfrws_police.pcap
SHA256	1837ee390e060079fab1e17caff88a1837610ef951153ddcb7cd85ad478228e

## 2 Analysis Result Summary

Before explaining the details of the analysis, this chapter reconstructs the situation of the crime scene based on the facts found through the analysis, and presents the solution of the questions based on this.

### 2.1 Reconstruction of the Crime Scene

This section organizes the actions that have occurred in the Drug Lab on from May 15 to May 17.

Evidences underlying the speculation are given in part 3.3.2 Detailed Timeline (DT) or other specific section of the report.

The figure below summarizes the main acts taken place in the laboratory, and we describe them with more detail in the following paragraphs.

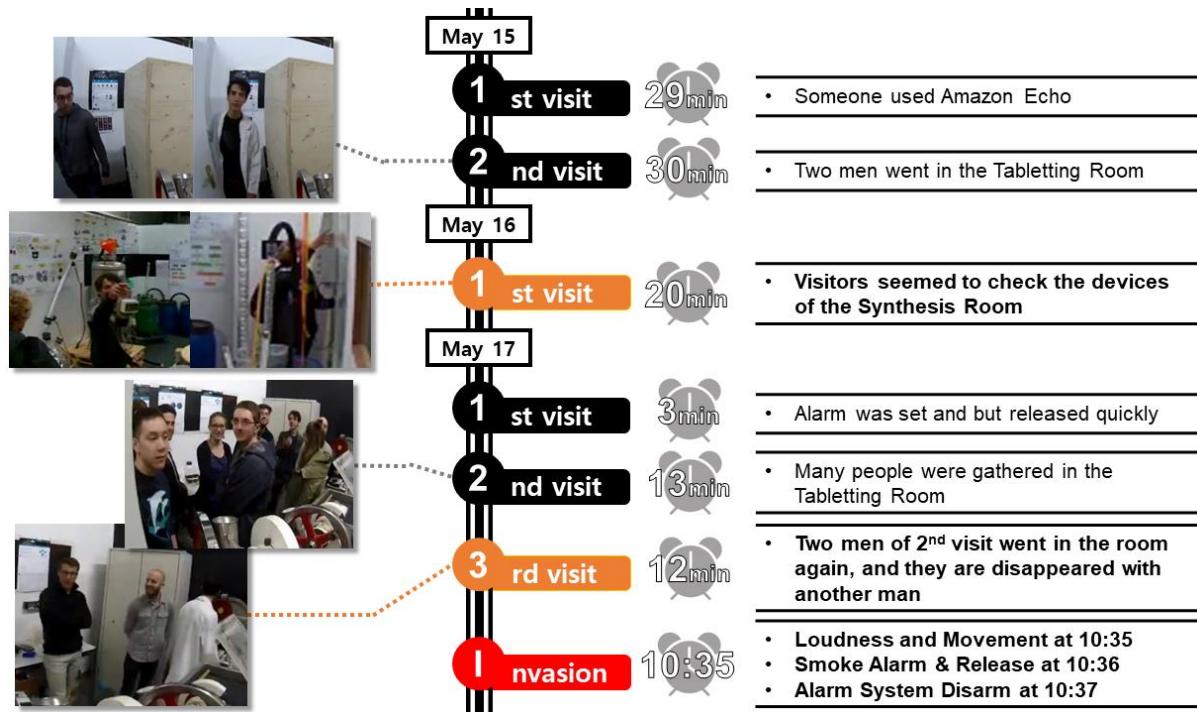


Figure 2 – The main acts taken place in the laboratory, from May 15 to May 17

#### May 15, 2018: Two Days Before The Raid

There would have been **two visits** to the lab on this day.

##### First Visit

- The first visit lasted for about 29 minutes from 11:10 to 11:39, and there is no other trail than trying to activate(arm) iSmartAlarm system with Amazon Echo. (DT No 15 – 27)
- At 14:38:20, an operator named 'JPinkman' activated the alarm system. (DT No. 28)

## Second Visit

- The second visit lasted about 30 minutes from 14:41 to 15:11. (DT No. 30 - 81)
- At 14:41:35, JPinkman deactivated(disarmed) the alarm system. (DT No. 31)
- The operators 'pandadodu' and 'TheBoss' manipulated the alarm system several times. (DT No. 41 – 47)
- **Two men went in and out of the tabletting room.** (DT No. 32 - 76)
- Amazon Echo was used for ordinary purposes from 15:05 to 15:07. (DT No. 53 – 74)



**Figure 3 – Two men who visited the tabletting room on May 15**

## May 16, 2018: One Day Before The Raid

On this day, there was **a visit for about 20 minutes** from 15:32 to 15:52. (DT No. 85 – 119)

- When the door of the lab was opened for the first time on this day, the alarm didn't work. This is because the alarm system was deactivated by JPinkman at 15:05:27 the day before. (DT No. 50)
- Amazon Echo was used for ordinary purposes from 15:34 to 15:36. (DT No. 89 – 105)
- At 15:33:55, the door was closed and then at 15:34:45 opened again. This is the time when Amazon Echo was being used, so it is possible to guess that someone other than the person who originally visited came in. (DT No. 88, 96)
- **Visitors were mainly in the synthesis room where Arlo Pro is installed in.** (DT No. 86, 87, 107, 108, 117, 118)
- **The people in the room seemed to check the camera and the smoke detector (Nest Protect).** (Cached Still Pictures of 'Arlo' and 'Wink – Smart Home' application in part 3.1.1.3 Application Data)



**Figure 4 – Still Pictures of the synthesis room on May 16 (remained in the cache directory of ‘Arlo’ and ‘Wink – Smart Home’ application)**

- At 15:55:27, JPinkman activated the alarm system. (DT No. 126)

**The reason why we insist that the still pictures taken by Arlo Pro were taken on this day are as follows.**

- Unlike ‘Nest’ application, ‘Arlo’ application does not store still pictures taken over time. Instead, some photos taken from the camera remain cached in ‘Arlo’ and ‘Wink - Smart Home’ application.
- Because same pictures are remained in cache directories of both applications, **the pictures would have been taken after May 16, 2018, 02:49:53**, the time when Arlo Pro was linked to the Wink Hub. (DT No. 83, 84)
- **The pictures would have been taken before May 17, 2018, 09:40**, the time when the caches were created. (DT No. 127, 128)
- Therefore, it is possible to guess that the still pictures remaining in the two applications were taken during the visit on May 16.

### **May 17, 2018: The Day of The Incident**

There would have been **three visits** to the lab on this day.

## First Visit

- The first visit lasted about 3 minutes from 09:44 to 09:47. (DT No. 129-132)
- The alarm worked when the door opened, but TheBoss released it.
- At 9:47:50 after the visit, JPinkman activated the alarm system.

## Second Visit

- The second visit lasted about 13 minutes from 10:09 to 10:22. (DT No. 133 – 156)
- The door was opened at 10:09:52 and TheBoss cleared the alarm. (DT No. 135)
- Someone is in the synthesis room from 10:10. **From 10:14 to 10:16, the large box in the tabletting room was removed and many people gathered in there.** During this time, there was an attempt to play music using Amazon Echo. (DT No. 136 – 149)



**Figure 5 – Many people who gathered in the tabletting room on the second visit of May 17**

- After people left, at 10:22 someone activated the alarm system using Amazon Echo. The door was not closed at this time. (DT No. 151 – 155)

## Third Visit

- The third visit lasted about 12 minutes from 10:22 to 10:34. (DT No. 157 – 174)
- Although we don't know when someone came in the lab exactly because the door had been opened, we can estimate approximate time as 10:22, by the fact that TheBoss deactivated the alarm system at 10:22:30 and Nest Cam detected loudness at this moment. (DT No. 157, 158)
- From 10:30 the two men entered the tabletting room and kept inside. They are who were on the second visit. (DT No. 162 – 167)



**Figure 6 – Two men came in the tabletting room again on the third visit of May 17**

- At 10:34:15, the door closes, and at 10:34:17, TheBoss sets the alarm system to the Home mode. In this mode, the alarm system only detects door opening and ignores internal movement. (DT No. 168, 169)
- **At 10:34:31, pandadodu deactivated the alarm system**, and the door of the lab was opened after 5 seconds.  
It is possible to assume that **this operator would be the person who entered the tabletting room at 10:34:39.** (DT No. 170 – 172)



**Figure 7 – Another man (maybe ‘pandadodu’) came in the tabletting room at 10:34:39**

- Short time after man who is assumed to be pandadodu came in, the tabletting room was empty from 10:34:52. (DT No. 174)

#### **After The Visits (Invasion)**

- At 10:35:32, Nest Cam detected loudness, and at 10:35:53, Arlo Pro detected movement, so someone might have entered the synthesis room at about this time. (DT No. 175, 176)
- In less than a minute, Nest Protect detected smoke at 10:36:06. However, the alarm was lifted at 10:36:20 and only last 14 seconds. (DT No. 177 – 180)

- At 10:37:52, **pandadodu deactivated the alarm system**, and at 10:38:52, motion was detected in the synthesis room. (DT No. 181, 182)
- At 10:40, the police recognized the incident and at 10:45 the forensic team arrived at the scene. (DT No. 183, 184)

## 2.2 Answers to the Challenge Questions

### 1) At what time was the illegal drug lab raided?

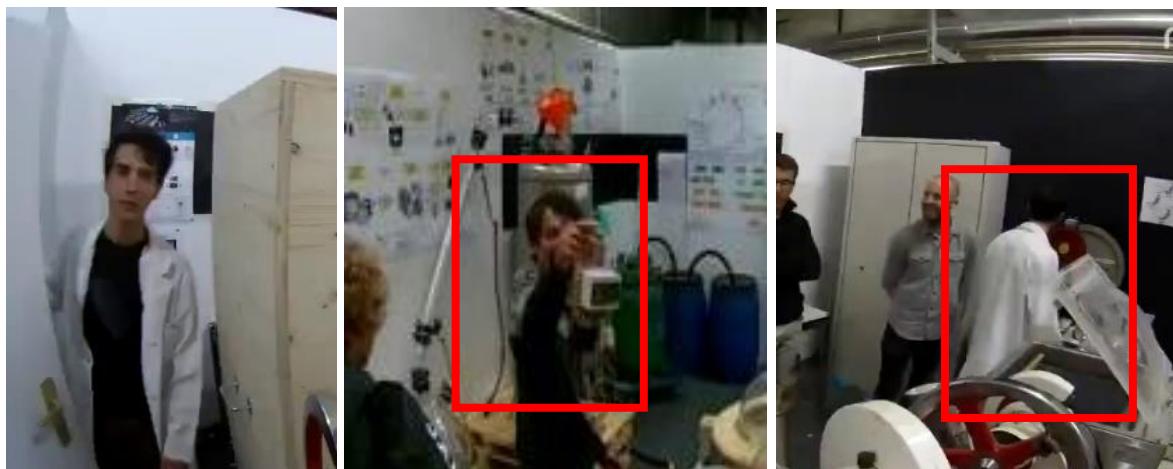
**Answer : Approximately at 10:35.**

- When we look at what happened on the day of the incident, it can be assumed that **the raid occurred between 10:34:52, the time that the tabletting room was empty, and 10:36:06 when Nest Protect sensed the smoke.**
- Narrowing the time horizon, we can insist that **the intruder(s) entered the lab near at 10:35:32, when Nest Cam detected the sound.**

### 2) Could any of the two friends of Jessie Pinkman have been involved in the raid?

**Answer : Yes, D.Pandana could have.**

- **From May 15 to May 17**, several people appear in the still photographs taken by Nest Cam and Arlo Pro. Of them, **a man continues to appear, as shown in the picture below.**



**Figure 8 – A man who continuously appeared in the lab from May 15 to May 17**

- The man is supposed to be D. Pandana because **the operator 'pandadodu' deactivated the alarm system shortly before the man entered the tabletting room at 10:34 on the day of the incident.** The nickname 'pandadodu' is probably a combination of the first and last names of him.

- At the time of the raid, iSmartAlarm system was deactivated and Nest Protect was released only in 14 seconds after detecting the smoke. This implies that there is a high possibility that insider or someone who acquired information through insider who knew about the security system of the lab was involved in the invasion.
- People who appeared on May 16 did some strange behavior, such as pointing the camera in the synthesis room or touching Nest Protect.  
If these people are not members of the lab, they can be suspicious of carrying out recon of the security system in advance before entering the lab, and Pandana, who was with them at this time, is likely to be an insider.
- Even if they were members of the lab, there is still something strange about his behavior. At about 10:34 on the day of the incident, when he entered the tabletting room, the room was empty after a short time and smoke was detected soon. It may be that he left the room with fellows to let the lab be penetrated by the intruder(s).
- In addition, the fact that the alarm system was deactivated by him, in about a minute after the smoke alarm was turned off, is a suspicious part of the incident.
- Therefore, police will need to investigate further facts, including whether Pandana uses 'pandadodu' as nickname at the iSmartAlarm system.

### 3) How was the QBee camera disabled?

**Answer : A vulnerability of the camera could be used to disable the camera without permission.**

- In the Drug Lab's network, Jessie Pinkman's smartphone had sent heartbeat packets to QBee Camera for streaming session. These packets contain authentication cookies unencrypted and anyone who can sniff them can reuse the cookies to execute arbitrary commands like switching to private mode.
- Given that Pandana and Varga admit having access to the network, they are able to sniff the network, to exploit the vulnerability and to make the camera disabled.
- Considering the answer of 2nd question, Pandana may be involved in the deactivation of the camera.
- For more explanation about the vulnerability, refer to part 3.2.2 Conversations.

### 3 Forensic Analysis Details

This chapter describes the process of analyzing the given data and the details of the findings. First, we look at the data extracted from each device, and then at the data collected on the network. At the end, we associate individual facts to deduce additional information and organize facts chronologically to cover a whole range of behaviors that took place in the crime scene.

Tools used for the analysis are summarized in the table below, and newly developed tools for the analysis are described in Chapter 4.

**Table 2 – Tools used for digital forensic analysis**

Name	Version	Usage
Autopsy [1]	4.9.1	File system Browsing
Binwalk [2]	2.1.2	Binary Image Component Analysis
DB Browser for SQLite [3]	3.9.1	SQLite Database Browsing
Wireshark [4]	3.0.0	Network Packet Analysis
010 Editor [5]	7.0.2	Raw Data Viewer
Realm Studio [6]	3.5.0	Realm Database Browsing
Nest Home Simulator [7]	2.2.1	Simulating Environment of Nest Devices
Online JSON Viewer [8]	-	Beautifying JSON Data
FFmpeg [9]	3.2.12	Video Data Recovery
Python [10]	3.7.2	Running & Developing Tools with Python Script
Docker [11]	17.12.0-ce	Application Container Platform
Elasticsearch [12]	6.5	Data Search Engine
Kibana [13]	6.5	Visualization Engine & User Interface
dex2jar [14]	2.0	Android Application Decomplier

The hash values of the individual files that appeared during the analysis and some additional or detailed explanation of the analysis are attached to the appendix.

#### 3.1 Device Level Analysis

This section describes the key information found in acquired data extracted from devices at the scene of the incident.

##### 3.1.1 Jessie Pinkman's Smartphone

Jessie Pinkman's smartphone, discovered at the Drug Lab, is an Android operating system based smartphone, Samsung Galaxy S6 Edge (SM-G925F). Key specification of this device [15] is as follows.

**Table 3 - Specification of Jessie Pinkman's smartphone**

Specification of SM-G925F (Samsung Galaxy S6 Edge)	
Dimensions	143.4 x 70.5 x 6.8mm
Weight	138g
Operating System	Android 5.0.2 Lollipop
Size (Main Display)	5.1" (129.2mm)
Resolution (Main Display)	QHD 2560 x 1440
Processor	Exynos 7420 (Quad-Core 1.5GHz & Quad-Core 2.1GHz)

RAM Size	3GB
ROM Size	32GB, 64GB, 128GB
Wi-Fi	802.11 a/b/g/n/ac 2.4G+5GHz, VHT80 MIMO
Bluetooth	Version 4.1
Battery Capacity	2550mAh
Release Date	2015, April

The acquired data is provided as a compressed file, and there are four files in the compressed file.

- proCDATA.zip: CPU information, storage related settings, event log, etc.
- **blk0\_sda.bin: The main storage space of the device.** It includes hardware-related setting partition, a recovery partition, and a user data partition that stores applications, media files, and so on.
- blk16\_sdb.bin, blk32\_sdc.bin: It is assumed to be the data area associated with the boot loader called 'sboot'. The basis for this conjecture is described in the appendix.

No significant information related to the incident could be found in the files except for 'blk0\_sda.bin', which may leave traces of the user. Therefore, **we have analyzed 'blk0\_sda.bin' intensively, and all of what we explain below are found in this file.**

When navigating the 'blk0\_sda.bin' file through Autopsy, **18 volumes (partitions) are recognized** as listed in the table below. **When referring to the location of the file, if we do not specify a volume name, it means that the file belongs to the 'USERDATA' volume.** Otherwise, we specify the volume name like '/[vol EFS]/'.

**Table 4 – Name of recognized volumes in 'blk0\_sda.bin'**

BOTA0	BOTA1	EFS	PARAM
BOOT	RECOVERY	OTA	RADIO
TOMBSTONES	DNT	PERSISTENT	STEADY
PERSDATA	SBFS	SYSTEM	CACHE
HIDDEN	USERDATA		

### 3.1.1.1 Basic Information

**Table 5 – Basic information of Jessie Pinkman's smartphone**

Path	Information	
/[vol EFS]/wifi/.mac.info	MAC Address of Wi-Fi NIC	AC:5F:3E:73:E3:78
/[vol EFS]/bluetooth/bt_addr or /misc/bluedroid/bt_config.conf	Bluetooth Address	D8:C4:E9:7C:2E:F8
/property/persist.sys.timezone	Timezone	Europe/Zurich
/misc/wifi/wpa_supplicant.conf	Saved Wi-Fi AP	SSID "Cthulhuuuu's iPhone" SSID "ESC-IoT"

The following are additional descriptions of the information in the table above:

- Timezone: 'Europe/Zurich' is originally UTC+1, but Daylight Saving Time (DST) was applied from March 25 to October 28, 2018, indicating that it was UTC+2 at the time of the incident [16].
- Saved Wi-Fi AP

- SSID “Cthulhuuu's iPhone”
  - ◆ PSK: 289c0fa13f7bb57468d69ca0f092caeef
  - ◆ Security Method: WPA-PSK
  - ◆ Frequency: 2412MHz
- SSID “ESC-IoT”
  - ◆ PSK: 5af1a71868975ebcea2be5ac5f7ea71d
  - ◆ Security Method: WPA-PSK
  - ◆ Frequency: 2412MHz

### 3.1.1.2 Media File

Before analyzing application data installed on the smartphone, we first looked at the areas that users can easily access while using the smartphone. This enabled us to **obtain information related to crime scenes from several photo files.**

#### 1) Camera Directory

Photographs taken by user are stored in ‘/media/0/DCIM/Camera’.

In this directory, **we can find some pictures that show the model name, MAC address, etc. of various devices found in the Drug Lab.** Those pictures were taken at from March 26 to April 10, 2018.



**Figure 9 – Photograph of the Arlo base station**

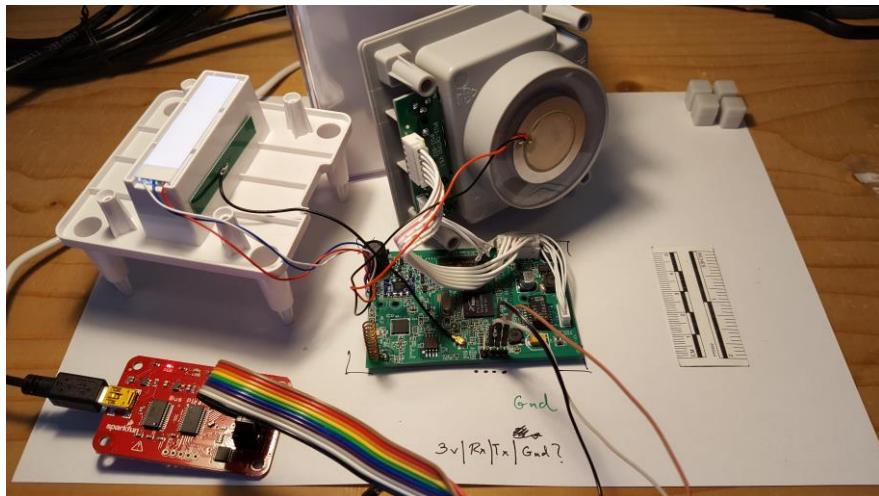
The device information that can be grasped through photographs is summarized in the following table.

**Table 6 – Device information identified through Photographs in ‘Camera’ directory**

iSmartAlarm Base Station (through ‘20180410_092015.jpg’)	
Model Name	iPU3G
MAC Address	00:4D:32:09:D9:E4
iSmartAlarm Motion Sensor (through ‘20180410_092059.jpg’)	
Model Name	PIR3G
Serial Number	141605015143012
Arlo Base Station (through ‘20180326_164922.jpg’)	
Model Name	VMB4000

MAC Address	08:02:8E:FF:75:4F
Serial Number	4RD37B75A1EC9
Wink Hub (through '20180410_091838.jpg')	
MAC Address	B4:79:A7:25:02:FA
Serial Number	16170011WZD1
Nest Protect (through '20180410_091924.jpg')	
Model Number	A14
Network Name	NEST-E345
Entry Key	157PT9
Serial Number	06CA01AC331600CA
Nest Cam (through '20180410_092120.jpg')	
Model Number	A0005
MAC Address	18:B4:30:61:C9:EF
Entry Key	FUJY0N

Between April 10 and 25, scenes of disassembled device or of approach to UART(Universal Asynchronous Receiver/Transmitter) port were photographed.



**Figure 10 - Photograph of UART approach to iSmartAlarm base station**

The following table lists the filenames associated with the scene.

**Table 7 – Photographs of disassembly & UART approach**

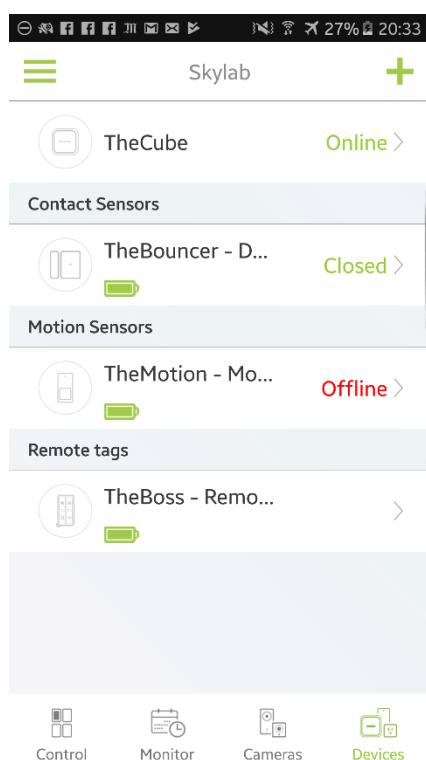
Description	Filename	
Arlo Base Station Disassembly	20180410_114114.jpg	20180410_114127.jpg
	20180410_114136.jpg	20180410_152808.jpg
Arlo Base Station UART Approach	20180410_114908.jpg	20180410_114913.jpg
iSmartAlarm Base Station Disassembly	20180425_164054.jpg	20180425_164335.jpg
	20180425_164540.jpg	20180425_164621.jpg
	20180425_165004.jpg	20180425_165025.jpg
	20180425_165051.jpg	
iSmartAlarm Base Station UART Approach	20180425_173708.jpg	20180425_181145.jpg
	20180425_181201.jpg	20180425_181209.jpg

## 2) Screenshot Directory

Screenshots of smartphone display are stored in '/media/0/DCIM/Screenshots'.

Through a picture named 'Screenshot\_20180330-203354.png' in this directory, we can see the screen of application used to control 'iSmartAlarm' alarm system. By this picture, we got to know the existence and nickname of each component of alarm system.

- iSmartAlarm System at 'Skylab' (guessed to mean the Drug Lab)
  - Base Station '**TheCube**'
  - Contact Sensor '**TheBouncer**'
  - Motion Sensor '**TheMotion**'
  - Remote Tag '**TheBoss**' (remotely arm or disarm the alarm system)



**Figure 11 – Screenshot of 'iSmartAlarm' application**

The 'Screenshot\_20180502-132904.png' file provides information on the MAC address, software version, etc. of QBee Camera. The table below summarizes the identifiable device information of the camera.

**Table 8 – Device information of QBee camera identified through screenshot**

QBee Camera (through 'Screenshot_20180502-132904.png')	
MAC Address	D8:FB:5E:E1:01:92
S/W Version	4.16.4
H/W Version	4.1
Serial Number	416B4067717
WLAN Info	Connected at 'ESC-IoT' (Signal Strength 75%)

### 3.1.1.3 Application Data

Most of the task on smartphones is done through applications, and each application stores the user's information, settings, usage logs, and sometimes creates a temporary file like cache. Depending on the type of application, the type and contents of the data to be stored will be different. We can analyze the data remaining in the various applications used by the smartphone user, so that we can grasp various information related to the user.

Application data is usually stored in a subdirectory of '/data/data/', named after application ID, and sometimes also in a subdirectory of '/media/0/Android/data/' or in a subdirectory of '/media/0/' with a specific name.

Reviewing Pinkman's smartphone application data, we could not find meaningful information in contacts, messages, SNS (Google Hangout, Facebook, WhatsApp, etc.) and many other basic applications. Instead **there were many traces left in applications for managing devices installed in the Drug Lab.**

#### 1) iSmartAlarm (iSA.common)

##### ● Alarm System History

The 'iSmartAlarm.DB' file is located in '/data/iSA.common/databases/' directory.

It is SQLite database and contains 24 tables. The main table in the database is 'TB\_IPUDairy'.

25	1526544597		004D3209D9E4	2		TheBoss				2	DISARM
26	1526545342		004D3209D9E4	2		JPinkman				0	ARM
27	1526545350		004D3209D9E4	2		TheBoss				2	DISARM
28	1526546057		004D3209D9E4	2		TheBoss				1	HOME
29	1526546071		004D3209D9E4	2		pandadodu				2	DISARM
30	1526546272		004D3209D9E4	2		pandadodu				2	DISARM

**Figure 12 – Table 'TB\_IPUDairy' of 'iSmartAlarm.DB' in 'iSmartAlarm' application**

This table contains the alarm mode (profile) change history. A record in the table consists of timestamp, ID of IPU (base station), operator, and alarm mode.

There are 4 types of alarm mode in iSmartAlarm system [17]:

- **ARM** (0) - Contact sensors and motion sensors are activated.  
By default, you're given 3 minutes to leave your home before the system is armed.
- **HOME** (1) - Only contact sensors are activated.  
Movements occurred indoor will not trigger the alarm.  
By default, Home mode immediately arms the system.
- **DISARM** (2) - The system is immediately deactivated and no longer armed.
- **PANIC** (3) - The siren is activated immediately and all members will immediately receive a notification indicating the member who set it off.

The following table lists the alarm mode change history recorded in 'TB\_IPUDairy' table. ('IPUID' field of all records have value '004D3209D9E4', which indicates the base station of the Drug Lab.)

**Table 9 – Alarm mode change history recorded in 'iSmartAlarm' application**

Date	Operator	Profile (Alarm Mode)
2018-05-15 14:38:20	JPinkman	ARM
2018-05-15 14:41:35	JPinkman	DISARM

Date	Operator	Profile (Alarm Mode)
2018-05-15 15:03:11	pandalodu	PANIC
2018-05-15 15:03:28	pandalodu	PANIC
2018-05-15 15:03:36	pandalodu	DISARM
2018-05-15 15:03:43	pandalodu	PANIC
2018-05-15 15:03:47	TheBoss	DISARM
2018-05-15 15:04:43	pandalodu	HOME
2018-05-15 15:05:27	JPinkman	DISARM
2018-05-16 15:47:14	TheBoss	PANIC
2018-05-16 15:47:15	TheBoss	PANIC
2018-05-16 15:47:18	TheBoss	HOME
2018-05-16 15:47:33	TheBoss	DISARM
2018-05-16 15:47:40	TheBoss	DISARM
2018-05-16 15:53:10	JPinkman	HOME
2018-05-16 15:53:31	JPinkman	ARM
2018-05-16 15:55:07	JPinkman	DISARM
2018-05-16 15:55:11	JPinkman	DISARM
2018-05-16 15:55:17	JPinkman	DISARM
2018-05-16 15:55:27	JPinkman	ARM
<b>2018-05-17 09:45:22</b>	<b>TheBoss</b>	<b>DISARM</b>
<b>2018-05-17 09:47:50</b>	<b>JPinkman</b>	<b>ARM</b>
<b>2018-05-17 10:09:57</b>	<b>TheBoss</b>	<b>DISARM</b>
<b>2018-05-17 10:22:22</b>	<b>JPinkman</b>	<b>ARM</b>
<b>2018-05-17 10:22:30</b>	<b>TheBoss</b>	<b>DISARM</b>
<b>2018-05-17 10:34:17</b>	<b>TheBoss</b>	<b>HOME</b>
<b>2018-05-17 10:34:31</b>	<b>pandalodu</b>	<b>DISARM</b>
<b>2018-05-17 10:37:52</b>	<b>pandalodu</b>	<b>DISARM</b>

As you can see in the above table, there are 3 operators appeared in the records.

- **JPinkman**
- **TheBoss**
- **pandalodu**
  
- User Information

The '**iSmartAlermData.xml**' file is located in '/data/iSA.common/shared\_prefs/' directory, and contains information such as the ID and password that are used to connect to the cloud service.

The vulnerability (CVE-2018-16222) is present in version 2.0.8 of the application [18]. this vulnerability makes anyone who can access to this XML file be able to see the login credentials and passwords for the cloud service as cleartext.

This files shows that **Pinkman has used ID '0792245315' and password 'esc\_iot\_2018' to connect to iSmartAlarm's cloud service.**

```

1  <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2  <map>
3      <string name='password">esc_iot_2018</string>
4      <string name='phoneNum">0792245315</string>
5      <string name='monitorShowFlag'>fragment3_6_monitoring</string>
6  </map>

```

Figure 13 – Contents of ‘iSmartAlermData.xml’

## 2) Wink – Smart Home (com.quirky.android.wink.wink)

- Device Information and Events

The ‘**persistenceDB**’ file is located in ‘/data/com.quirky.android.wink.wink/databases/’ directory.

It is SQLite database and **table ‘Elements’ contains a lot of records about devices associated with the hub and activities (events) occurred from those devices**. Important information in the record is contained in ‘Json’ field with JSON format.

	ID	Type	Json	Name
1	a278021c-6c7...	local_hub	{"mHubData":null,"mId":"421391","mKeyHash":"B7:DB:5A:AC:59:4A:12:B9:C5:E7:46:3C:09:B5:21:70:35:E1:96:C1:19:C1:54:E8:C4:C0:71:A5:E5:66:CD:DA","mOAuth":{"access_token":"a-VqLJPY...}}	NULL
2	e480e766-644...	activity	{"action":{},"action_automation_mode":null,"action_id":null,"action_name":null,"action_type":null,"object_id":null,"...}	NULL
3	815083	linked_service	{"account":null,"credentials":null,"facets":[]}, "invalidated_at":null,"linked_service_id":815083,"relink_method": "none","service": "google_now","third_party": null}	NULL
4	1044132	linked_service	{"account": "f8896851a03a1cd6ad84f0a4f23f774f196cee8e","credentials":null,"facets":[]}, "invalidated_at":null,"linked_service_id":1044132,"relink_method": "oauth","service": "nest","third_party": null}	NULL
5	1044134	linked_service	{"account":null,"credentials":null,"facets":[]}, "invalidated_at":null,"linked_service_id":1044134,"relink_method": "provisioning_flow","service": "ama...}	NULL
6	43fd4242-e9de-4934-9...	activity	{"action":{},"action_automation_mode":null,"action_id":null,"action_name":null,"action_type":null,"object_id":null,"...}	NULL
7	3a1a445b-a3f8-4934-ae...	activity	{"action":{},"action_automation_mode":null,"action_id":null,"action_name":null,"action_type":null,"object_id":null,"...}	NULL

Figure 14 – Table ‘Elements’ of ‘persistenceDB’ in ‘Wink – Smart Home’ application

Type of information contained in a record can be distinguished by ‘Type’ field. There are various kinds of values appeared in this field, but only some types have significant information as described below:

- Type ‘hub’: Wink Hub device information

There is only one record (ID: 421391) that has ‘hub’ type.

Major information found in ‘Json’ field of the record is summarized below.

Table 10 – Major information of Wink Hub, found in ‘persistenceDB’

Key	Value
uuid	fa608f62-a137-4745-beb5-210fcfd6721af
object_id	421391
device_manufacturer	wink
manufacturer_device_model	wink_hub
user_ids	470654
last_reading::mac_address	B4:79:A7:25:02:FA
last_reading::firmware_version	2.66.0
last_reading::ip_address	10.20.30.22

- Type ‘camera’, ‘smoke\_detector’, and ‘light\_bulb’: associated device information

There are two records (ID: 237267, 235946) that have ‘camera’ type, one record (ID: 212474) that has ‘smoke\_detector’ type, and two records (ID: 1889042, 1700816) that have ‘light\_bulb’ type.

The following table summarizes basic information of the associated devices, based on those records.

**Table 11 – Basic information of associated devices of Wink Hub in ‘persistenceDB’**

Key	Value
Camera (ID 235946)	
created_at	1526374420 (2018-05-15 08:53:40 UTC)
device_manufacturer	nest
manufacturer_device_id	fEj8x-PFe3i4k9MauZ21gZB4q_a4CvdO7D-4t8QO_-dMcddUKG22vw
model_name	Nest Cam
name	SuperLab Tabletting Camera
Camera (ID 237267)	
created_at	1526431793 (2018-05-16 00:49:53 UTC)
device_manufacturer	netgear
manufacturer_device_id	arlo:59U17B7BB8B46
model_name	Arlo Pro
name	Kitchen's camera
Smoke Detector (ID 212474)	
created_at	1526374419 (2018-05-15 08:53:39 UTC)
device_manufacturer	nest
manufacturer_device_id	VpXN4GQ7MUDNDjAjS-6y80xx11Qobba_
model_name	Protect
name	SuperLab Kitchen Nest Protect (LabSmoker)
Light Bulb (ID 1700816)	
created_at	1463343015 (2016-05-15 20:10:15 UTC)
device_manufacturer	cree
manufacturer_device_id	-
model_name	Cree light bulb
name	Piano
Light Bulb (ID 1889042)	
created_at	1469704691 (2016-07-28 11:18:11 UTC)
device_manufacturer	cree
manufacturer_device_id	-
model_name	Cree light bulb
name	Upstairs

- Type ‘activity’: events from devices associated with the hub

There are 65 records that have ‘activity’ type. **JSON contents of each record has ‘object’ key that contains ID and name of device which made the event and it is possible to distinguish which event occurred on which device.**

The following table lists events recorded in the database, identified by activity type records.

All keys are included in ‘Json’ field. timestamp is from ‘created\_at’ key; device column is from ‘object’

key; event column is from ‘reading’ key.

**Table 12 – Events of the associated devices of Wink Hub recorded in ‘persistenceDB’**

Date	Device	Event
2018-05-15 10:53:39	Nest Protect	Smoke and CO not detected
2018-05-15 10:53:40	Nest Cam	Capturing video on
2018-05-15 11:12:15	Nest Cam	Capturing video off
2018-05-15 11:17:29	Nest Cam	Capturing video on
2018-05-15 11:33:39	Nest Cam	Capturing video off
2018-05-15 11:38:56	Nest Cam	Capturing video on and motion detected
2018-05-15 14:41:10	Nest Cam	Motion detected
2018-05-15 14:51:21	Nest Cam	Motion detected
2018-05-15 14:59:11	Nest Cam	Motion detected
2018-05-15 14:59:51	Nest Cam	Motion detected
2018-05-15 15:00:54	Nest Cam	Motion detected
2018-05-15 15:05:41	Nest Cam	Motion detected
2018-05-15 15:10:57	Nest Cam	Motion detected
2018-05-15 15:16:00	Nest Cam	Motion detected
2018-05-16 02:49:55	Arlo Pro	Armed
2018-05-16 15:32:46	Arlo Pro	Motion detected
2018-05-16 15:33:33	Arlo Pro	Motion detected
2018-05-16 15:34:49	Nest Cam	Loudness detected
2018-05-16 15:35:10	Nest Cam	Loudness detected
2018-05-16 15:35:24	Nest Cam	Loudness detected
2018-05-16 15:36:27	Nest Cam	Loudness detected
2018-05-16 15:37:35	Arlo Pro	Motion detected
2018-05-16 15:39:55	Arlo Pro	Motion detected
2018-05-16 15:47:53	Nest Cam	Loudness detected
2018-05-16 15:49:54	Arlo Pro	Motion detected
2018-05-16 15:50:13	Arlo Pro	Motion detected
2018-05-16 15:52:45	Nest Cam	Motion detected
2018-05-17 10:10:11	Arlo Pro	Motion detected
2018-05-17 10:10:45	Arlo Pro	Motion detected
2018-05-17 10:14:05	Nest Cam	Motion detected
2018-05-17 10:14:12	Arlo Pro	Motion detected
2018-05-17 10:15:00	Nest Cam	Motion detected
2018-05-17 10:15:56	Arlo Pro	Motion detected
2018-05-17 10:15:58	Nest Cam	Motion detected
2018-05-17 10:16:59	Nest Cam	Motion detected
2018-05-17 10:22:36	Nest Cam	Loudness detected
2018-05-17 10:28:50	Arlo Pro	Motion detected
2018-05-17 10:30:20	Arlo Pro	Motion detected
2018-05-17 10:30:30	Nest Cam	Motion detected
2018-05-17 10:30:50	Nest Cam	Motion detected
2018-05-17 10:31:48	Nest Cam	Motion detected
2018-05-17 10:32:50	Nest Cam	Motion detected

Date	Device	Event
2018-05-17 10:33:15	Nest Cam	Motion detected
2018-05-17 10:33:37	Nest Cam	Motion detected
2018-05-17 10:34:44	Nest Cam	Motion detected
2018-05-17 10:35:32	Nest Cam	Loudness detected
2018-05-17 10:35:53	Arlo Pro	Motion detected
<b>2018-05-17 10:36:06</b>	<b>Nest Protect</b>	<b>Smoke detected</b>
<b>2018-05-17 10:36:20</b>	<b>Nest Protect</b>	<b>Smoke not detected</b>
2018-05-17 10:38:52	Arlo Pro	Motion detected

We can see that the cameras detected movements several times, and the Nest Protect detected smoke and was disarmed quickly on the day of the incident.

The ‘wink\_local\_pref\_470654.xml’ file is located in ‘/data/com.quirky.android.wink.wink/shared\_prefs/’ directory.

Like what we found in ‘persistenceDB’, we can see the IDs of devices connected to Wink Hub, grouped by ‘camera’, ‘smoke\_detector’, ‘hub’, ‘light\_bulb’, etc.

```

1 ?xml version='1.0' encoding='utf-8' standalone='yes' ?
2 <map>
3   <string name="order:camera">[&quot;camera/237267&quot;,&quot;camera/235946&quot;]</string>
4   <string name="order:group">[&quot;group/5412465&quot;,&quot;group/5412466&quot;,&quot;group/8014846&quot;,&quot;group/5412469&quot;,&quot;
5   <string name="order:smoke_detector">[&quot;smoke_detector/212474&quot;]</string>
6   <string name="feature_flags">[(&quot;conditions&quot;:[(&quot;active_after&quot;:&null,&quot;active_before&quot;:&null,&quot;hidden_at&quot;:
7     <string name="order:hub">[&quot;hub/4213916&quot;]</string>
8   <string name="favorite_pref">[]</string>
9   <boolean name="add_hub_later" value="false" />
10  <string name="order:light_bulb">[&quot;light_bulb/1889042&quot;,&quot;light_bulb/1700816&quot;,&quot;group/5412467&quot;]</string>
11  <string name="order:sensor_pod">[]</string>
12 </map>
13
14

```

Figure 15 – Contents of ‘wink\_local\_pref\_470654.xml’

- Cached Still Pictures

In ‘/data/com.quirky.android.wink.wink/cache/image\_manager\_disk\_cache/’ directory, we can see JPEG image caches of still pictures taken by the Nest Cam and Arlo Pro.

Because those caches are made by the application, their creation time would not match with the time the images were taken.

So we don’t know when the scene of a still picture really happened, but by the fact that **cache files from Nest Cam were created at May 15, 2018, from 15:10:07 to 15:10:24**, and files from **Arlo Pro were created at May 17, 2018, from 09:40:31 to 09:40:37**, we can insist that they were taken at the time before their creation time. Also, mere witnessing the presence of some scenes would be enough to us to figure out what happened in the Drug Lab.

The following table is a collection of important photos of the synthesis room and the tabletting room found in this directory.

**Table 13 – Important still pictures remained in cache directory of ‘Wink – Smart Home’ application**

Tabletting Room (Taken by Nest Cam)	
3bf47cf7de82576cc12107fd27867f9a847991ab6 bf36c98aaa124e1db888e3a.0	ce801688848e08931fd5f5ca8597671920929098 97d2eeeea59390d0773092b84.0
	
A man is walking around the room.	
Synthesis Room (Taken by Arlo Pro)	
e6decba9336ebb90a64c84579b5be16ebdf6303 408f1a2432d5cf4dba16e1e3e.0	4bf6c92039987221a8e7a28614e5a216241d143 523535c05651be1977bd90db6.0
	
A man is pointing at the camera with his finger.	Another man is touching a device. According to the diagram of the Drug Lab, the device would be Nest Protect.

- Login Account

You can see that both ‘user.xml’ file and ‘com.quirky.android.wink.wink\_preferences.xml’ file in ‘/data/com.quirky.android.wink/shared\_prefs/’ directory contain string ‘emidnight@gmail.com’, which may have been used to login account.

```

1  <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2  <map>
3    <string name="authuser">&quot;clearTokens&quot;;false,&quot;email&quot;:&quot;emidnight@gmail.com&quot;,&quot;first_name&quot;:
4      <string name="2e4af5373f018c2285e7c92b9889a05">mIt0J8o3tl0=
5    </string>
6    <boolean name="isauth" value="true" />
7    <string name="@0b8cf9f2700508f158fcdd572b478f">LMA0+2YYs7JJJo+UCYCPg6Y8RC0lybZAwC1DRxsAA1pTfiV69oG0fcEosj0cXuuJRX/cfunE3t9x
8      3d50qcM48P16hGw8AwzmapT/41jaZ1oZ2o9e4puS05XTf0f/puYsXVXSVdSg5SW25EbJSdz74ooB
9      axjVC6X1PDjyWSSHGm0lZ2qrDTvJRaoZ27LSf10X0ATm2V8NCwRzAMP5tYlgTTRZ+bSBvb6sA6nFo
10

```

**Figure 16 - Contents of ‘user.xml’ in ‘Wink – Smart Home’ application**

```

1  <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2  <map>
3    <int name="APP_LAUNCHES" value="6" />
4    <int name="mobile_id" value="2636015" />
5    <string name="com.quirky.android.wink.core.login.pref.EMAIL">emidnight@gmail.com</string>
6
7  </map>
8

```

**Figure 17 - Contents of ‘com.quirky.android.wink.wink\_preferences.xml’**

### 3) Nest (com.nest.android)

- Recorded Video Frames

The ‘frame\_database’ file is located in ‘/data/com.nest.android/cache/f315c6e2b5434a5381f1f5be6f73b4b3/’ directory and is SQLite database.

**‘frame\_raw\_data\_table’ table in this database contains video frame data recorded by Nest Cam, encoded with RAW H.264 codec.** Each record in the table contains a video frame with timestamp, and videos – concatenation of successive still pictures – can be extracted by combining 20 frame records that have same ‘chunk\_id’ field value. The data structure of the table is described in detail in the appendix.

By analyzing the data structure of the table, we made a parser to extract individual frames or videos from the database and can see what happened in the tabletting room where the Nest Cam was installed.

We extracted 228 videos (4548 frames) which contain still pictures taken from May 14 to May 17, 2018.

Most of the videos show only empty tabletting room with light turned off, but **some videos show the visiting of some people in the room**. They are listed in the following table. (Index is zero-based, and frame is one-based.)

**Table 14 – Notable video files extracted from ‘frame\_database’ in ‘Nest’ application**

Index	Chunk ID	Time Range
78	1526388330	2018-05-15 14:45:28 - 2018-05-15 15:05:16
79	1526389528	2018-05-15 15:05:27 - 2018-05-15 15:26:30
201	1526543298	2018-05-17 09:48:16 - 2018-05-17 10:14:00
202	1526544850	2018-05-17 10:14:43 - 2018-05-17 10:34:01
203	1526546051	2018-05-17 10:34:27 - 2018-05-17 10:55:21

- Index 78 and 79

These videos contain the scene where **two men go around the tabletting room on May 15**.

The following table summarizes what we can see in the videos.

**Table 15 – Summary of video 78 and 79 extracted from ‘frame\_database’**

2018-05-15 14:51:08 (frame 6 of video 78)	2018-05-15 14:51:18 (frame 7 of video 78)
	
It seems that someone is coming in this room. (we can see a hand holding the door)	A man A is going out.

2018-05-15 14:58:55 (frame 12 of video 78)	2018-05-15 15:00:38 (frame 17 of video 78)
	
Another man B is coming in the room.	the man B is going out.
2018-05-15 15:03:01 (frame 18 of video 78)	2018-05-15 15:04:08 (frame 19 of video 78)
	 A red box highlights the door handle area.
The room is empty.	The door opened slightly.
2018-05-15 15:05:27 (frame 1 of video 79)	2018-05-15 15:06:40 (frame 2 of video 79)
	
the man B appeared again.	The room is empty again with the door closed.
2018-05-15 15:09:06 (frame 4 of video 79)	2018-05-15 15:10:20 (frame 5 of video 79)
	
the man A appeared again.	The room is empty again (3 <sup>rd</sup> ) with the door closed.

■ Index 201, 202 and 203

These videos contain the scene where **many people come and go in the tabletting room at the time before the raid occurred.**

The following table summarizes what we can see in the videos.

**Table 16 – Summary of videos between 201 and 203 extracted from ‘frame\_database’**

2018-05-17 10:14:00 (frame 20 of video 201)	2018-05-17 10:14:43 (frame 1 of video 202)
	
Someone pushes a box toward the outside of the room.	People are coming in the room.
2018-05-17 10:15:21 (frame 4 of video 202)	2018-05-17 10:16:41 (frame 8 of video 202)
	
Many people gather in the room.	people are going outside.
2018-05-17 10:22:25 (frame 9 of video 202)	2018-05-17 10:30:36 (frame 13 of video 202)
	
The room is empty.	Two of them come in the room again.
2018-05-17 10:34:39 (frame 2 of video 203)	2018-05-17 10:34:52 (frame 3 of video 203)
	
Another man comes into the room.	The room is empty again.

- **Device Events**

There is an SQLite database named ‘cache’ in ‘/data/data/com.nest.android/databases/’ directory. ‘entries’ table of the database stores information about cached data by key as the below figure.

cache	2018-05-15 13:35:50 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:58:47 GMT+02:00		
cache-journal	2018-05-15 13:35:50 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00		
<					
Hex Strings Application Indexed Text Message File Metadata Results Annotations Other Occurrences					
Table entries 2 entries Page 1 of 1 ← → Export to CSV					
key	path	created	last_modified	last Accessed	policy
bucketMap	cache-1332523362.json	1526384150412	1526578364142	1526384150412	0
account	cache-1503821048.json	1526578364129	1526578364129	1526578364129	0

**Figure 18 – Table ‘entries’ of ‘cache’ in ‘Nest’ application**

The cache file ‘cache-1332523362.json’ corresponding to the ‘bucketMap’ key is located in ‘/data/com.nest.android/cache/cache/’ directory and has JSON format. **It contains event records of registered Nest devices.**

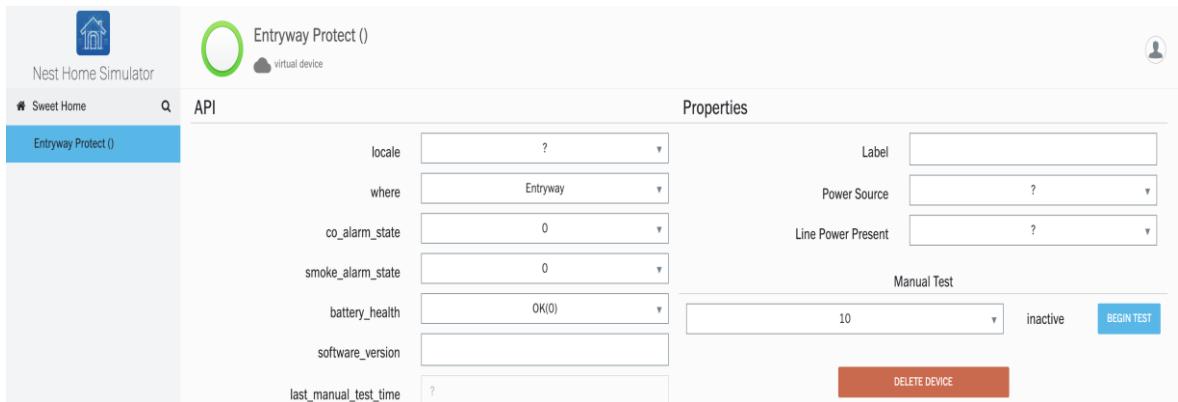
This file contains 17 objects, and each object consists of 4 keys, which are ‘object\_key’, ‘object\_revision’, ‘object\_timestamp’ and ‘value’. ‘object\_key’ and ‘value’ keys must exist in all of elements but ‘object\_revision’ and ‘object\_timestamp’ keys may not exist.

The following table enumerates ‘object\_key’ key of each element.

**Table 17 – List of object in ‘cache-1332523362.json’**

Index	object_key
0	widget_track.18B430000039E345
1	where.a3e36480-5757-11e8-80bb-0e2d565eed46
2	structure.a3e36480-5757-11e8-80bb-0e2d565eed46
3	topaz_resource.5
4	user.9201208
5	utility.a3e36480-5757-11e8-80bb-0e2d565eed46
6	partner_programs.a3e36480-5757-11e8-80bb-0e2d565eed46
7	delayed_topaz.18B430000039E345
8	safety.a3e36480-5757-11e8-80bb-0e2d565eed46
9	<b>message_center</b> .9201208
10	structure_metadata.a3e36480-5757-11e8-80bb-0e2d565eed46
11	user_settings.9201208
12	structure_history.a3e36480-5757-11e8-80bb-0e2d565eed46
13	geofence_info.a3e36480-5757-11e8-80bb-0e2d565eed46
14	entitlements.a3e36480-5757-11e8-80bb-0e2d565eed46
15	quartz.f315c6e2b5434a5381f1f5be6f73b4b3
16	topaz.18B430000039E345

To find out how an event occurred by Nest devices is recorded in the cache when a fire actually occurred, we created a virtual environment using ‘Nest Home Simulator’. we also installed ‘Nest’ application on the Android virtual machine to see what traces are left when we change the status of ‘co\_alarm\_state’ and ‘smoke\_alarm\_state’.



**Figure 19 – Nest Home Simulator**

After installing the application, we changed the smoke detection state to 1 in the Nest Home Simulator. When an event occurred, a message was notified in the application.

Through this test, we found that event message objects are stored at ‘messages’ array in ‘value’ key of the 9<sup>th</sup> object, object\_key of which is ‘message\_center.\*’, with the structure as the following table shows.

**Table 18 – Structure of message object stored in ‘cache-1332523362.json’**

Key	Description	Example
thread_id	Thread ID of message (same ID for same message)	1526546166676
read	Whether or not the user has acknowledged the message	False
priority	Priority (the lower the number, the higher the priority)	1
timestamp	The time the message arrived (the time the event occurred)	1526546180 (12018-05-17 08:36:20 UTC)
dismissed	Whether message is delivered	True
key	message type	protect_smoke_warn_clear
id	message unique ID	5fab5b80-59ad-11e8-bce3-12f5f322bc9e
parameters	message content (device name, location, time zone, etc.)	'18B430000039E345', '00000000-0000-0000-00010000000a', 'Kitchen', 'LabSmoker', 'SuperLab', 0, 0, 0, 2, 1, 1, 'a3e36480-5757-11e8-80bb-0e2d565eed46', 'Europe/Zurich'

The following tables are event messages generated by Nest Protect and stored in the cache file.

**Table 19 – Event messages generated by Nest Protect and stored in ‘message\_center’ object**

Thread ID	Date	Type of Event
0	2018-03-16 12:08:45	protect_power_out_now
0	2018-03-16 14:24:18	protect_power_out_now
0	2018-04-09 09:38:23	protect_power_out_now
0	2018-04-27 13:12:16	protect_power_out_now
0	2018-04-27 15:59:06	protect_power_out_now

0	2018-04-29 10:50:51	protect_power_out_now
0	2018-05-02 07:36:45	protect_power_out_now
1526546166676	<b>2018-05-17 10:36:06</b>	<b>protect_smoke_warn</b>
1526546166676	<b>2018-05-17 10:36:20</b>	<b>protect_smoke_warn_clear</b>

The above data shows that the actual fire occurred and Nest Protect detected it at May 17, 2018, 10:36:06 and after 14 seconds of the detection the smoke alarm was released.

- Still Pictures in Communication Caches

The '/data/com.nest.android/cache/dcnetwork' directory is presumed to have http communication data cached.

📁 [parent folder]	2018-05-15 13:16:56 GMT+02:00	2018-05-15 15:12:55 GMT+02:00	2018-05-15 15:12:55 GMT+02:00	2018-05-15 13:16:56 GMT+02:00
📁 [current folder]	2018-05-15 13:35:13 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 13:35:13 GMT+02:00
📁 -532597445-1277665155	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00
📁 -532597445-1563280601	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00
📁 -532597445-949917945	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00	2018-05-15 15:12:58 GMT+02:00

Figure 20 – File list of ‘dcnetwork’ directory in ‘Nest’ application

Communication target host is ‘nexusapi-eu1.dropcam.com’. In those caches, we can extract JPEG format images one by one from the three files. **Extraction can be done through carving header signature and footer signature of the JPEG format from the files.**

02C0h:	00 66 39 39	66 34 37 63	62 2D 32 35	34 31 2D 34	.f99f47cb-2541-4
02D0h:	34 62 62 2D	38 30 64 30	2D 66 37 32	35 33 35 38	4bb-80d0-f725358
02E0h:	35 35 38 35	37 FF D8 FF E0	00 10 4A	46 49 46 00	55857ÿØÿà..JFIF.
02F0h:	01 02 00 00	01 00 01 00	00 FF DB 00	43 00 03 02	.....ÿÛ.C...
0300h:	02 03 02 02	03 03 03 03	04 03 03 04	05 08 05 05	.......
0310h:	04 04 05 0A	07 07 06 08	0C 0A 0C 0C	0B 0A 0B 0B	.....
0320h:	0D 0E 12 10	0D 0E 11 0E	0B 0B 10 16	10 11 13 14	.....
0330h:	15 15 15 0C	0F 17 18 16	14 18 12 14	15 14 FF DB	.....ÿÛ
0340h:	00 43 01 03	04 04 05 04	05 09 05 05	09 14 0D 0B	.C.....

Figure 21 – JPEG header signature found in cache file in ‘dcnetwork’ directory



Figure 22 – JPEG image carved at the cache file in ‘dcnetwork’ directory

- User Information

The ‘com.nest.android.preferences.xml’ file is located in ‘/data/com.nest.android/shared\_prefs/’ directory. Here we can find user e-mail account of the application, ‘jpinkman2018@gmail.com’.

The cache file ‘cache-1503821048.json’ is located in ‘/data/com.nest.android/cache/cache/’ directory and has JSON format. It contains user e-mail account of the application, ‘jpinkman2018@gmail.com’.

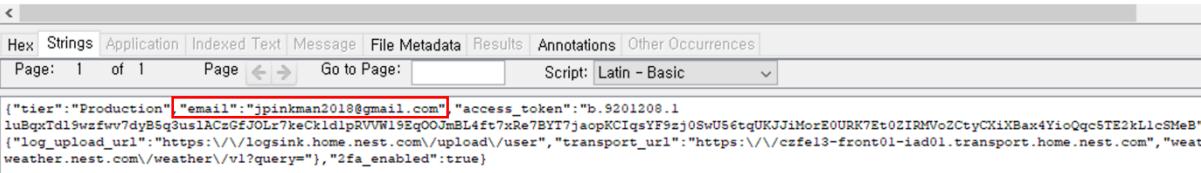
```

1 <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2 <map>
3   <string name="dropcam_token">2040aadc1448498ab22a8ed134974f5efb208d90039aecb06797c802f24a2fe4b18a6bfe</string>
4   <string name="userLogin">jpinkman2018@gmail.com</string>
5   <string name="account_snapshot_transport_url">https://czfe13-front01-iad01.transport.home.nest.com</string>
6   <boolean name="snapshot_load_from" value="true" />
7   <string name="tier_id">Production</string>
8   <string name="user_id">9201208</string>
9   <string name="user_token">b.9201208.1luBqxTdl9wzfwv7dyB5q3uslACzGfJ0Lr7keCk1d1pRVVV19Eq00JmBL4ft7xRe7BYT7jaopKCIqsYF9zj0SwL
10  </map>
11
12

```

**Figure 23 - Contents of ‘com.nest.android.preferences.xml’**

cache-1503821048.json	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00
tmp_cache-1332523362.json	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00	2018-05-17 19:32:44 GMT+02:00



```

{
  "tier": "Production",
  "email": "jpinkman2018@gmail.com",
  "access_token": "b.9201208.1luBqxTdl9wzfwv7dyB5q3uslACzGfJ0Lr7keCk1d1pRVVV19Eq00JmBL4ft7xRe7BYT7jaopKCIqsYF9zj0SwL",
  "log_upload_url": "https://\\logsink.home.nest.com\\upload\\user",
  "transport_url": "https://\\czfe13-front01-iad01.transport.home.nest.com",
  "weather.nest.com\\weather\\v1?query=",
  "2fa_enabled": true
}

```

**Figure 24 - Contents of ‘cache-1503821048.json’**

#### 4) Arlo (com.netgear.android)

- Cached Still Pictures

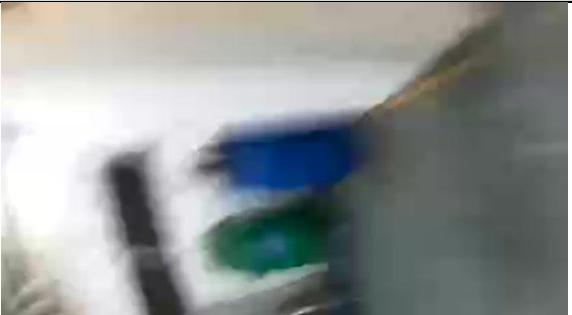
In ‘/data/com.netgear.android/cache/http/’ directory, we can see JPEG format caches of still pictures taken by the Arlo Pro, like what we see in ‘Wink – Smart Home’ application.

As we explained the reason in the part of Wink Hub application data, we don’t know when the scene of a still picture really happened, but we can insist that they were taken at the time before their creation time. The cache files were created at May 17, 2018, from 09:39:36 to 09:40:23.

Furthermore, mere witnessing the presence of some scenes would be enough to us to figure out what happened in the synthesis room of the Drug Lab.

The following table is a collection of important photos of the synthesis room found in this directory.

**Table 20 – Important still pictures of the synthesis room, remained in cache directory of ‘Arlo’ application**

67f43186731b8ca1b0dbf25ec25c5391.0	995a98df55fff0452ef26af1da49e80d.0
	
A man is pointing at the camera with his finger.	The camera seems to have shaken and lost focus.
a52213fe457b9b76431d909d1bce6d7d.0	41593f144685f25ba6494ce186d1ff7a.0
	
A man is touching a device. According to the diagram of the Drug Lab, the device would be Nest Protect.	
a03a80b969307da39f6fcb01a2d29af9.0	92af7d396df2aa692712d3cedc7ce004.0
	
A man puts something into a pot.	There are three man in the room.

- Device Information

The ‘**default.realm**’ file is located in ‘/data/com.netgear.android/files/’ directory and is Realm format [19] database. ‘**DatabaseModelDevicesData**’ class of the database contains a record about registered Arlo devices, in JSON format.

CLASSES	Enter a query to filter the list	Create DatabaseModelDevicesData
ArloAutomationConfigurationRealmObject	0	userid string (Primary Key)
DatabaseModelAccountData	1	A79GZN-316-31881729
DatabaseModelCamerasData	1	
DatabaseModelDeviceCapabilitiesData	2	
DatabaseModelDevicePermissionsData	0	
DatabaseModelDevicesData	1	devicesJSON string?
DatabaseModelE911LocationsData	1	[{"userId": "A79GZN-316-31881729", "deviceId": "4RD37B75A1EC9", "uniqueId": "A79GZN-316-31881729_4RD37B75A1EC9", "deviceType": "basestation"}]

**Figure 25 – Record of ‘DatabaseModelDevicesData’ class in ‘default.realm’**

The value of ‘devicesJSON’ field of the record beautified by JSON viewer online are as the following figure.

0	<ul style="list-style-type: none"> <li>- userid : "A79GZN-316-31881729"</li> <li>- deviceId : "4RD37B75A1EC9"</li> <li>- uniqueId : "A79GZN-316-31881729_4RD37B75A1EC9#1"</li> <li>- deviceType : "siren"</li> <li>- deviceName : "Superlab GW"</li> <li>- xCloudId : "CSTYXBN-1690-316-47062328"</li> <li>- userRole : "OWNER"</li> <li>- displayOrder : 2</li> <li>- mediaObjectCount : 0</li> <li>- state : "provisioned"</li> <li>- modelId : "VMB4000-siren"</li> <li>- dateCreated : 1526289276189</li> <li>- owner <ul style="list-style-type: none"> <li>- firstName : "Jessie"</li> <li>- lastName : "Pinkman"</li> <li>- ownerId : "A79GZN-316-31881729"</li> </ul> </li> <li>- properties <ul style="list-style-type: none"> <li>- modelId : "VMB4000"</li> <li>- olsonTimeZone : "Europe/Amsterdam"</li> <li>- hwVersion : "VMB4000r3"</li> </ul> </li> </ul>	1
2	<ul style="list-style-type: none"> <li>- userid : "A79GZN-316-31881729"</li> <li>- deviceId : "59U17B7BB8B46"</li> <li>- parentId : "4RD37B75A1EC9"</li> <li>- uniqueId : "A79GZN-316-31881729_59U17B7BB8B46#1"</li> <li>- deviceType : "camera"</li> <li>- deviceName : "Kitchen's camera"</li> <li>- lastModified : 1526578366312</li> <li>- xCloudId : "CSTYXBN-1690-316-47062328"</li> <li>- lastImageUploaded : "true"</li> <li>- userRole : "OWNER"</li> <li>- displayOrder : 1</li> <li>- presignedLastImageUrl : "https://arlolastimage-z1.s3.amazonaws.com/arlo/mobile/1690-316-47062328/59U17B7BB8B46/lastimage.jpg?Expires=1526578366&amp;Signature=..."</li> <li>- presignedSnapshotUrl : "https://arlo3-prod-z1.s3.amazonaws.com/arlo/mobile/1690-316-47062328/59U17B7BB8B46/snapshot.jpg?Expires=1526578366&amp;Signature=..."</li> <li>- presignedFullFrameSnapshotUrl : "https://arlo3-prod-z1.s3.amazonaws.com/arlo/mobile/1690-316-47062328/59U17B7BB8B46/fullframe.jpg?Expires=1526578366&amp;Signature=..."</li> <li>- mediaObjectCount : 8</li> <li>- state : "provisioned"</li> <li>- modelId : "VMC4030"</li> <li>- dateCreated : 1521034297853</li> <li>- arloMobilePlan : false</li> <li>- interfaceVersion : "1003"</li> <li>- interfaceSchemaVer : "3"</li> <li>- owner <ul style="list-style-type: none"> <li>- firstName : "Jessie "</li> <li>- lastName : "Pinkman"</li> <li>- ownerId : "A79GZN-316-31881729"</li> </ul> </li> <li>- properties <ul style="list-style-type: none"> <li>- modelId : "VMC4030"</li> <li>- olsonTimeZone : "Europe/Amsterdam"</li> <li>- hwVersion : "H11"</li> </ul> </li> </ul>	3

**Figure 26 – Beautified JSON object originally contained in the record of ‘DatabaseModelDevicesData’ class**

You can see device information of the Arlo base station, the camera (Arlo Pro) and the siren (integrated with base station).

The following table summarizes the major information contained in the JSON object.

**Table 21 – Registered Arlo devices in ‘Arlo’ application**

Key	Value
1 <sup>st</sup> Object (Arlo Base Station)	
deviceId	4RD37B75A1EC9
deviceType	basestation
deviceName	Superlab GW
firmwareVersion	1.10.0.4_20963
modelId	VMB4000
dateCreated	1526289276194 (2018-05-15 09:14:36.194 UTC)
properties::hwVersion	VMB4000r3
connectivity::type	ethernet
2 <sup>nd</sup> Object (Arlo Base Station Siren)	
deviceId	4RD37B75A1EC9

deviceType	siren
deviceName	Superlab GW
modelId	VMB4000-siren
dateCreated	1526289276189 (2018-05-15 09:14:36.189 UTC)
3 <sup>rd</sup> Object (Arlo Pro)	
deviceId	59U17B7BB8B46
parentId	4RD37B75A1EC9
deviceType	camera
deviceName	Kitchen's camera
modelId	VMC4030
dateCreated	1521034297853 (2018-03-14 13:31:37.853 UTC)
properties::hwVersion	H11

- User Information

The ‘Phoenix.xml’ file is located in ‘/data/com.netgear.android/shared\_prefs/’ directory, and contains information like user e-mail account.

```

1 <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2 <map>
3   <int name="passwordLength" value="10" />
4   <string name="vuezoneUrl">https://arlo.netgear.com/hmsweb</string>
5   <string name="token">ceEBZaUJva6vmnekKxH5NtTA/ZteKmNaaYEx5hVlt6jhWmemMI7ZVPrGhk5p9e5XDe0A0XZyAqZF
6   Bp/2XXSVob1+ZVM996ZVrE8dgMzDZ8/V35JymMMnSHvM4lbcDWpBVwHGmIgn0TsjLHo4opYWir4
7   gAke2iyUYpppljc05NbKoBLg00o2Gd8SnILemwcm1fV0bNxkmFtd+z07Nsp6zTplWjmPyawAhNA6W
8   iGFr8k=
9     </string>
10    <string name="userId">A79GZN-316-31881729</string>
11    <string name="59U17B7BB8B46/snapshot">https://arlos3-prod-z1.s3.amazonaws.com/119af05d_33c1_47f1_970f.
12    <boolean name="touchIDLoginUsedLast" value="false" />
13    <string name="59U17B7BB8B46">available</string>
14    <string name="email">jpinkman2018@gmail.com</string>
15    <boolean name="showTouchIDOnboarding" value="false" />
16    <string name="alwaysRemind">true</string>
17

```

Figure 27 - Contents of ‘Phoenix.xml’

## 5) QBee Cam (com.vestiacom.qbeecamera)

The ‘com.vestiacom.qbeecamera\_preferences.xml’ file is located in ‘/data/com.vestiacom.qbeecamera/shared\_prefs/’ directory and contains encrypted user information.

Normally we can’t know the contents of this file, but a vulnerability (CVE-2018-16223) make anyone who can access to this file be able to decrypt the user information stored in the file. The vulnerability occurs under version 1.0.5 of the ‘QBee Cam’ Android application, and the key reason for the vulnerability is that data used to derive the decryption key are hard-coded in the application [18].

```

<map>
  <string name="4Y6xz8byViS81N4VAY0ZOjYYOZa21Os9NklpMj2gEA">3g9oh9jar0icqnsi7vep6jls4t</string>
  <string name="DGPwuGi4LKfQX0YCwDxHtw">kcugM+KZSjL+3cBbZagBdw</string>
  <string name="AFEvat4bO5WkgsNn2BMR1Q">pGA1aMO3Xrpbr37ip81pQg</string>
</map>

```

Figure 28 – Encrypted contents of ‘com.vestiacom.qbeecamera\_preferences.xml’

In order to protect user credentials, the application uses the Secure Preferences library to enhance the confidentiality of the storage with the Shared Preferences format. But because of the customization of the library, the data used for encryption can be obtained from the application source. This can be confirmed directly by decompiling the APK.

```

private static byte[] m7382c(String str) {
    StringBuilder stringBuilder = new StringBuilder();
    stringBuilder.append(str.substring(0, str.length() / 2));
    stringBuilder.append("a!k@ES2,g86AX&D8vn2]");
    stringBuilder.append(str.substring(str.length() / 2));
    byte[] bArr = null;
    try {
        bArr = MessageDigest.getInstance("SHA256").digest(stringBuilder.toString().getBytes());
    } catch (NoSuchAlgorithmException e) {
        e.printStackTrace();
    }
    return bArr;
}

```

**Figure 29 – Decompiled code snippet of ‘QBee Cam’ application related to the Secure Preference Routines**

We can obtain a SHA-256 hash digest that is derived from the string which is the combination of the string written in the decompiled source code and one of the data value in the preference file. With using this hash as the decryption key, we can decrypt other values of the preference file. The decoded results are as follows.

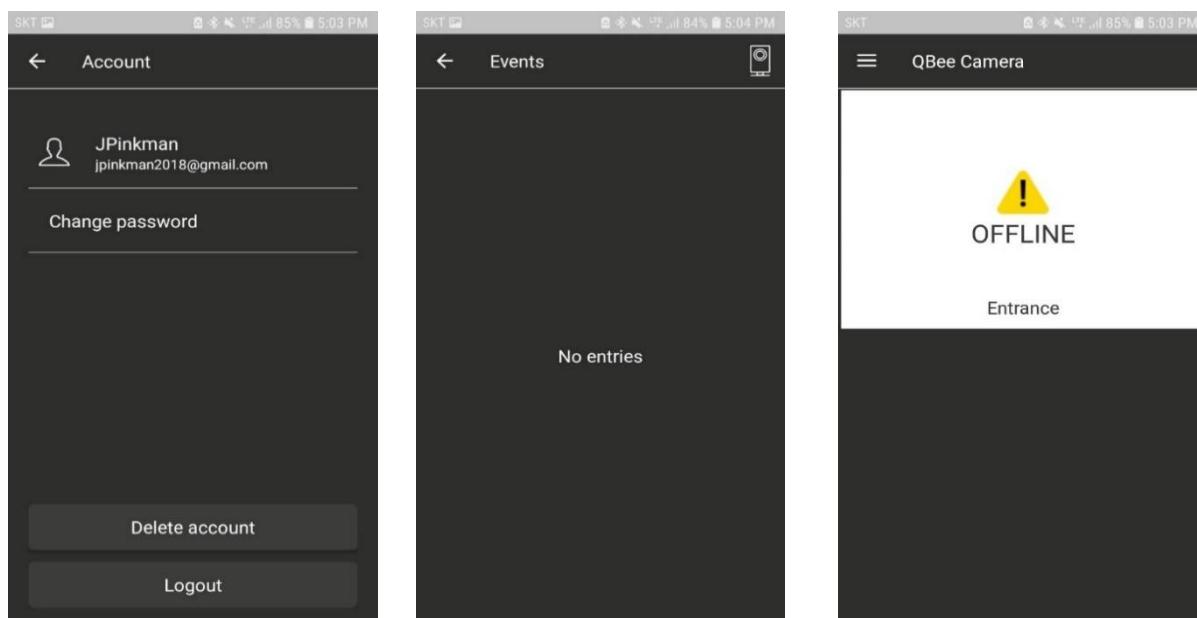
```

bmsinfo@extr:~/codes/ctf$ python crypto_dec.py com.vestiacom.qbeecamera_preferences.xml
{
    "decrypted_settings": [
        {
            "qbeePassword": "Esc_iot_2018",
            "qbeeUser": "JPinkman"
        }
    ]
}bmsinfo@extr:~/codes/ctf$ █

```

**Figure 30 – Decrypted user information from ‘com.vestiacom.qbeecamera\_preferences.xml’**

**Actually it was possible to access QBee cloud storage with this ID and password, but there was no record on the day of the incident.** This is because QBee stores videos only for up to two weeks for free app users. The following figure shows the cloud storage accessed with the Pinkman’s user account.



**Figure 31 – QBee Cam application access with the Pinkman’s user account**

## 6) Amazon Alexa (com.amazon.dee.app)

The ‘map\_data\_storage\_v2.db’ file located in ‘/data/com.amazon.dee.app/databases/’ directory contains user name in ‘accounts’ table and the ‘service.identity.xml’ file located in ‘/data/com.amazon.dee.app/shared\_prefs/’ directory contains user name and user e-mail address. The account username is ‘Jessie Pinkman’, and the e-mail address is ‘jpinkman2018@gmail.com’.

map_data_storage_v2.db	2018-05-15 13:25:22 GMT+02:00	2018-05-17 19:36:05 GMT+02:00	2018-05-17 19:58:48 GMT+02:00
map_data_storage_v2.db-journal	2018-05-15 13:25:22 GMT+02:00	2018-05-17 19:36:05 GMT+02:00	2018-05-17 19:36:05 GMT+02:00
Hex Strings Application Indexed Text Message File Metadata Results Annotations Other Occurrences			
Table accounts	1 entries	Page 1 of 1	Export to CSV
_id	directed_id	display_name	
amzn1.account.AGGMG4DRSURCQ7QT4TCLAINUZT2Q		Jessie Pinkman	

Figure 32 – User name stored in ‘map\_data\_storage\_v2.db’ in ‘Amazon Alexa’ application

```
b0      </set>
61      <string name="user.accessToken">Atna|EwICIHQl94wdlx4GyRv0Ny6fLXxX6geNKBRi
62      <string name="user.name">Jessie Pinkman</string>
63      <string name="user.[version]">5</string>
64      <string name="user.tokens">{}</string>
65      <string name="user.email">jpinkman2018@gmail.com</string>
66      <string name="user.profile.lastName">Pinkman</string>
   . . .
   . . .
```

Figure 33 – User name and e-mail address stored in ‘service.identity.xml’ in ‘Amazon Alexa’ application

With the user e-mail address and guessed password ‘esc\_iot\_2018’ which appeared in many other configuration files, we could log in Amazon Alexa management website [20] but couldn’t get in further because of the verification process.

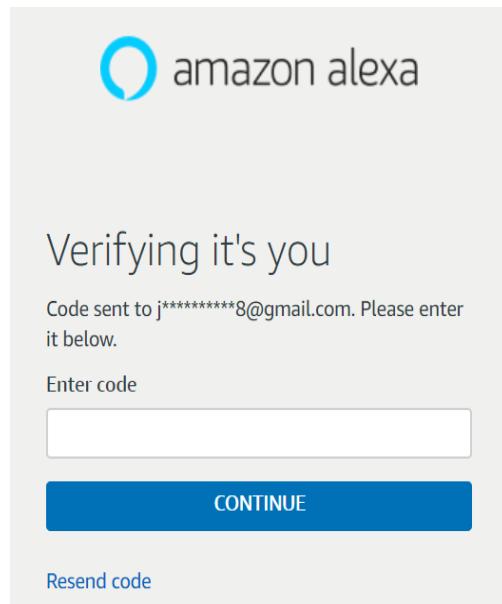


Figure 34 – Verification request after we log in Amazon Alexa management website

### 3.1.2 iSmartAlarm Base Station

**In the Drug Lab, sensors and base station made by iSmartAlarm are installed to monitor the entrance.** iSmartAlarm is a security system that detects intrusion into a building. The system consists of a base station and various sensors, and two types of sensor can be used. Motion sensors are to detect movement and contact sensors are to detect opening and closing of doors, windows and drawers. Those sensors inform users of detection information through a base station called 'CubeOne'.

Through internet searching and evidence, we found that the model name of base station that appeared in this scenario is 'iPU3G'. Key specifications of the base station [21] are as follows.

**Table 22 – Specification of iSmartAlarm base station**

Specification of iPU3G (iSmartAlarm CubeOne)	
Dimensions	100 x 100 x 105mm
Weight	370g
Maximum Working Distance	100m in open space
Network	10/100 BASE-T Ethernet (RJ-45 Connector)
Alarm Sound Level	Maximum of 110dB
Release Date	2015, April

There are two types of data obtained from this device.

- server\_stream: Diagnostic logs
- ismart\_00.img, ismart\_80.img: Physical memory dump image

#### 3.1.2.1 Diagnostic Logs

**In 'server\_stream' file, it is possible to see the detection records of motion sensors and contact sensors.** As the format of the file was not known, we developed a tool through basic analysis and are possible to interpret the contents recorded in the file.

The type of log record identified are divided into 'Unstructured Log' and 'Sensor Log', which are custom names for the convenience of research rather than official names. The Unstructured Log contains various values including the signature of the file, and the Sensor Log contains a record of the contact sensor and the PIR sensor (motion sensor).

The format of log record types we analyzed is attached to the appendix.

**The table below summarizes the sensor events recorded in the logs.**

**Table 23 – iSmartAlarm sensor events recorded in the diagnostic logs**

Time	Sensor Type	Event	Alarm
2018-05-14 11:33:07	Contact	Door Open	
2018-05-14 11:44:31	Contact	Door Closed	
2018-05-15 11:10:25	Contact	Door Open	
2018-05-15 11:39:03	Contact	Door Closed	
2018-05-15 14:41:28	Contact	Door Open	O
2018-05-15 15:04:57	Contact	Door Closed	
2018-05-15 15:04:59	Contact	Door Open	O
2018-05-15 15:11:05	Contact	Door Closed	
2018-05-15 15:11:06	Contact	Door Open	

2018-05-15 15:11:06	Contact	Door Closed	
2018-05-16 15:32:17	Contact	Door Open	
2018-05-16 15:33:55	Contact	Door Closed	
2018-05-16 15:34:45	Contact	Door Open	
2018-05-16 15:47:17	Contact	Door Closed	
2018-05-16 15:47:19	Contact	Door Open	O
2018-05-16 15:52:10	Contact	Door Closed	
2018-05-17 09:44:53	Contact	Door Open	O
2018-05-17 09:47:18	Contact	Door Closed	
2018-05-17 10:09:52	Contact	Door Open	O
2018-05-17 10:09:55	Motion	Motion Detected	O
2018-05-17 10:34:15	Contact	Door Closed	
2018-05-17 10:34:36	Contact	Door Open	

### 3.1.2.2 Memory Image

Although there are two memory image files, '**'ismart\_00.img'** and '**'ismart\_80.img'**', the two files have the same **SHA256 hash value** and the 'Compare Files' function in 010 Editor results in all bytes being identical when compared on byte-by-byte basis, so they are same file. Therefore, the analysis was conducted for 'ismart\_00.img' only.

First, we use Binwalk to check what data is contained in the memory image file. The result of executing the tool is summarized in the following table.

**Table 24 – Binwalk result of 'ismart\_00.img'**

Offset (Hexadecimal)	Description
0xF78	ulimage header, header size: 64 bytes, header CRC: 0x1A086BAF, created: 2013-04-28 09:57:43, image size: 107896 bytes, Data Address: 0x80200000, Entry Point: 0x80200000, data CRC: 0x8FDE24EE, OS: Linux, CPU: MIPS, image type: Standalone Program, compression type: none, image name: "SPI Flash Image"
0x215710	U-Boot version string, "U-Boot 1.1.3 (Apr 28 2013 - 17:57:40)"
0x215960	CRC32 polynomial table, little endian
0x300000	<b>LZMA compressed data</b> , properties: 0x5D, dictionary size: 33554432 bytes, uncompressed size: 6126127 bytes

Header of 'ulimage' at offset 0xF78 appears as a kernel for U-Boot (an open source bootloader for embedded devices), and we can guess that U-Boot exists, through the string at 0x215710 offset.

There is also a LZMA compressed data at offset 0x300000 which can be extracted and decompressed to get a file. We analyze this decompressed file with Binwalk and found another LZMA file at **0x440000** offset.

After decompressing the another LZMA file, we can get an archive in CPIO format. In the archive, there are many directories and files, which conform to the Linux file system structure, but we cannot find any significant information.

The following table lists the directory structure and major files of the archive file system.

**Table 25 - Contents of CPIO archive extracted from iSmartAlarm memory image**

Directory	Description
/bin/	Basic command executables including Busybox
/dev/	Providing access path to abstracted devices
/etc/	File system configuration ('fstab')
/etc_ro/	Read-only configuration files
/home/	Home directory for each account (no valid file)
/lib/	Libraries for executables (shared object)
/media/	Providing access path to media devices (no valid file)
/mnt/	Providing access path to mounted storage (no valid file)
/proc/	Providing access path to information related to active system and process
/sbin/	Command executables and shell scripts related to system control and configuration
/sys/	Providing access path to system parameter (no valid file)
/tmp/	Temporary files (no valid file)
/usr/	Auxiliary hierarchy for read-only user data (executables, architecture-independent files, etc.) iSmartAlarm default configuration ('share/iSmart/default_config')
/var/	Variable files (no valid file) (guessed to be symbolic-linked to other directories, such as '/tmp/')
/	Root directory

### 3.1.3 Arlo Base Station

**The Drug Lab is equipped with an Arlo wireless security camera and a base station to monitor the inside of the synthesis room.**

Through internet searching and evidence, we confirmed that the model name of base station in this lab is 'VMB4000', and key specifications of the base station [22] are as follows.

**Table 26 – Specification of Arlo base station**

Specification of VMB4000 (Arlo Base Station)	
Dimensions	215.9 x 57.15 x 165.1mm
Weight	110g
Processor	500Mhz MIPS 74 K processor
Memory	128MB Flash & 128MB RAM
Network	Ethernet & 2.4GHz Wi-Fi

There are three types of data obtained from this device.

- dfrws\_arlo.img: Physical memory dump image
- nvram.log: File that output device settings stored in NVRAM
- arlo\_nand.tar.gz: Configuration files, log files, etc. extracted from the '/tmp/media/nand' directory of the base station's active file system

#### 3.1.3.1 NVRAM Setting

It seems like that '**nvram.log**' shows settings of the base station saved in NVRAM and is acquired from the output of command 'nvram show' executed by active system.

**The setting file contains various items related to network, user, system parameter, etc., and the items are in the form of '[key]=[value]'.**

The following table summarizes the major information found in the file. 'Reference Lines' indicates which lines in the file contain information and 'Key' shows the keys of corresponding reference lines.

**Table 27 – Settings of Arlo base station found in NVRAM setting**

Reference Lines (Key)		Setting
546 (wl_ssid) 573 (wla_preset_ssid) 663 (wla_ssid)	Wi-Fi AP SSID	NTGR_VMB_1526289002
225 (wla_secu_type) 1302 (wla_temp_secu_type)	Wi-Fi Security Method	WPA2-PSK
423 (wl_wpa_psk) 553 (wl0_wpa_psk) 632 (wla_preset_passphrase) 1013 (wla_passphrase)	Wi-Fi AP WPA PSK	55B6BAA31C58FA339E32BE25AE332BF1EA1F 09F0D0FE5A620A7EE5D650B7B7E3EBB2F385 656D4DA757DC924F1D88AD6C3392E12066A6 F9C9C902E1CF90D9B70200000000000000000000 0000000000000000
934 (wan_ipaddr) 1231 (dhcp_offer_ip)	Allocated IP Address	10.20.30.17
224 (wan_gateway) 600 (dhcp_serv_ip)	Gateway IP Address	10.20.30.1
74 (et0macaddr) 521 (lan_hwaddr) 856 (sb/1/macaddr)	MAC Address (LAN)	08:02:8E:FF:75:4E
146 (wl0.1_hwaddr) 356 (wan_hwaddr2) 687 (wan_hwaddr) 799 (wan0_hwaddr2) 939 (macaddr) 1051 (wan0_hwaddr)	MAC Address (WAN)	08:02:8E:FF:75:4F
539 (board_id)	Board ID	U12H348T00_NETGEAR
977 (os_version)	OS Version	9.10.178.4002

### 3.1.3.2 Memory Image

In the memory image file, **we could not find any important information.**

Instead, we found some facts through an intensive analysis of the image file, which can give **some speculation about the image itself as below.**

- The memory image was acquired by shell interface.
- At offset 0x180FCCC, we found a command-line string, that contains the name of the memory image file and seems to be used for physical memory acquisition.

180:FCA0h:	C4 FC 80 01	A8 63 F3 00	C4 FC 80 01	CC FC 80 01	Äü€."có.Äü€.iü€.
180:FCB0h:	04 FE 80 01	B4 FC 80 01	B4 FC 80 01	A8 ED F5 00	.p€.'ü€.'ü€.'ið.
180:FCC0h:	04 FE 80 01	E8 30 F3 00	00 00 00 00	73 61 76 65	.p€.è06.....save
180:FCD0h:	20 31 30 2E	32 30 2E	33 30 2E	31 3A 64 66	10.20.30.1:dfrw
180:FCE0h:	73 5F 61 72	6C 6F 2E	69 6D 67	20 30 78 30	s_arlo.img 0x000
180:FCF0h:	30 30 30 30	30 20 30	78 31 30 30	30 30 30 30	00000 0x10000000
180:FD00h:	00 67 77 3D	31 30 2E	32 30 2E	33 30 2E	.gw=10.20.30.1..
180:FD10h:	7C FD 80 01	00 00 00 00	3C FD 80 01	C0 6C F5 00	ý€....<ý€.Àlð.
180:FD20h:	90 FD 80 01	40 DF 00 01	01 00 00 00	40 DE 00 01	.ý€.ØB.....ØB..

Figure 35 – Command-line string seems to be used for physical memory acquisition

- The command could be executed in active system by shell interface. Terminal protocol like telnet or SSH or serial interface approached from UART port – like what we see in the photographs taken by Pinkman’s smartphone - would provide the shell.
- The memory image seems to be the concatenation of the flash memory and the RAM.
- As we mentioned in specification, the base station has 128MB flash memory and 128MB RAM.
- The size of the image file is 256MB, and many bytes in both the front 128MB of the file and the back 128MB – including the first 4 bytes that appear to act as the signature - of it are same.
- Assuming that we split it the front 128MB and the back 128MB, differences between the front and the back are found only from offset 0xFE9120 ~ 0x1036A5D and 0x180F6F8 ~ 0x180FA0F.
- By this phenomenon, we can suppose that the image is the concatenation of the flash memory and the RAM which loaded the contents of the flash memory and is modified by active system.
- Many bits of the image seem to be flipped by unknown reason.
- We found many strings with some characters misspelled. the figure below shows one of the misspelled strings. String ‘incomplete w2ite’ in the highlighted area (offset 0x481684 of the file) must have been ‘incomplete write’.

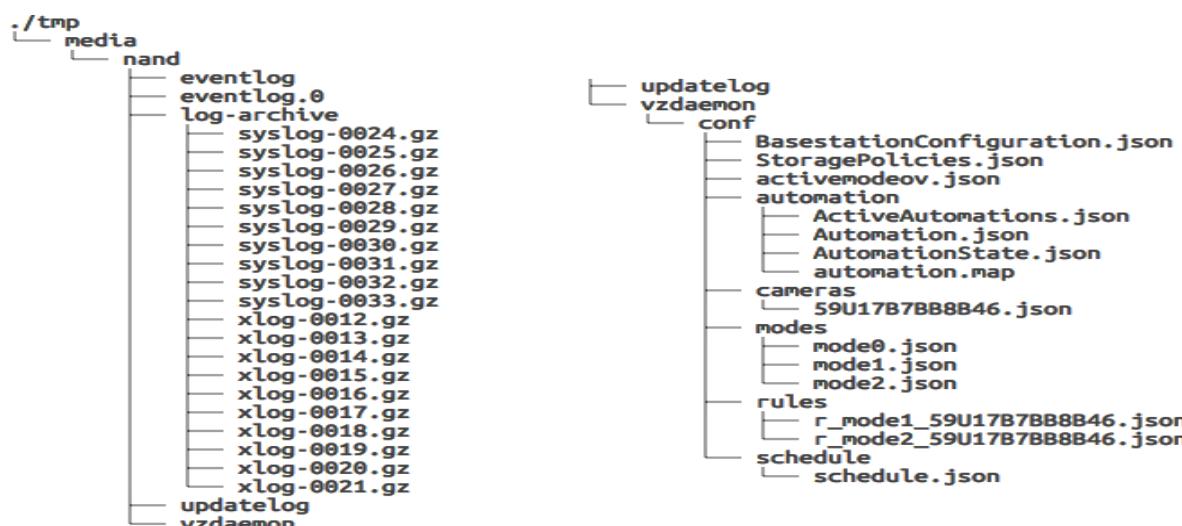
48:1670h:	20 64 61 F4	61 00 00 00	3C 31 3E 56	C1 4D 44 49	daða...<1>VÁMDI
48:1680h:	73 4B 3A 20	69 6E 63 6F	6D 70 6C 65	74 65 20 77	sK: incomplete w
48:1690h:	32 69 74 65	20 A8 25 64	20 21 3D 20	25 E4 29 08	2ite "%d != %ä).
48:16A0h:	00 00 00 04	2F 64 65 76	2F 72 61 6D	00 00 00 00	....../dev/ram....

Figure 36 – Irregularly transformed string found in the memory image

- We also found many areas where originally seem to be null-filled or 0xFF-filled have irregularly set bits or unset bits.
- Consequently, we lost many valuable data because Binwalk cannot identify or restore these flipped bits automatically. Even if we analyze it manually, we can just read some strings barely, and cannot restore data exactly, especially compressed files, as we don’t know where bit flips occurred.
- What’s strange is that these bit flipping occurred the same at the front 128MB and the back 128MB.

### 3.1.3.3 NAND TAR Archive

The file 'arlo\_nand.tar.gz' contains '/tmp/media/nand' directory of the base station's active file system. The following figure shows the structure of directories and files in this archive.



**Figure 37 – Directory tree of NAND TAR archive**

Although no significant information related to the incident was found in the archive, **it was possible to identify some facts about the base station** as in the following table.

**Table 28 – Some facts found in files of NAND TAR archive**

Path (in '/tmp/media/nand/')	Description
eventlog	IP Allocation (10.20.30.17)
vzdaemon/conf/automation/automation.map	You can see that 'mode0' means 'disarmed' and 'mode1' means 'armed'.
vzdaemon/conf/cameras/59U17B7BB8B46.json	Camera Information
vzdaemon/conf/BasestationConfiguration.json	Base Station Configuration AP Identifier: B8:27:EB:0E:3B:45 Timezone: Europe/Amsterdam (UTC+2)

The major information about camera device found in '59U17B7BB8B46.json' file is as below.

**Table 29 – Camera device information found in the configuration file**

Specification of Arlo Pro (through '59U17B7BB8B46.json')	
Model Name	VMC4030
ID	59U17B7BB8B46
H/W Revision	H11
S/W Version	1.092.0.13_19715
MAC Address	08:02:8E:FD:BD:CD
Resolution	1280 X 720
Streaming Mode	Event Based

### 3.1.4 Wink Hub

**In the Drug Lab, Wink Hub, which can control various IoT devices, such as bulbs, sensors, etc., is installed.** Previously, we found that the Arlo Pro, Nest Cam and Nest Protect (smoke detector) are connected to the hub.

Through internet searching and evidence, we found that the model name of hub that appeared in this scenario is 'PWHUB-WH01'. Key specifications of the hub [23] [24] are as follows.

**Table 30 – Specification of Wink Hub**

Specification of PWHUB-WH01 (Wink Hub)	
Dimensions	203 x 203 x 76mm
Operating System	Linux 2.6.35.3 flex-dvt
Bootloader	U-Boot 2014.01-14400 gda781c6-dirty
Processor	Freescale i.MX28 @454MHz (ARM926EJ-S)
RAM	64MB (DRAM)
Flash Memory	128MB NAND
Wi-Fi	802.11 b/g/n 2.4G
Supported Protocols	Bluetooth, Z-Wave (Security Enabled Z-Wave Plus Device), ZigBee, Wi-Fi, Lutron Clear Connect, Kidde
Supported Device Types	Z-WAVE: Door Lock, Door/Window Sensor, Motion Sensor, Tilt Sensor, Light Switch, Light Dimmer, Appliance Module, ZIGBEE: Light Bulb, Light Switch, Outlet, Door/Window
FCC Approval Date	2014, May

On this device, files and directories contained in the root file system were extracted. **The directory structure of the root file system is similar to the basic Filesystem Hierarchy Standard (FHS) of Linux**, and the following table lists the directory structure and major files.

**Table 31 – Structure of Wink Hub file system**

Directory	Description
/bin/	Basic command executables including Busybox
/database/	Linked device database ('apron.db', 'lutron-db.sqlite', etc.) Configuration files for Bluetooth, Wi-Fi, Zigbee Error log ('error.log') & Upgrade log ('upgrade_log')
/database_default/	Default database file
/dev/	Providing access path to abstracted devices
/etc/	Configuration files including network, file system, account information
/home/	Home directory for each account (no valid file)
/lib/	Libraries for executables (shared object) Firmware for various devices ('firmware/' directory)
/media/	Providing access path to media devices (no valid file)
/mfgtests/	Scripts for manufacturing test
/mnt/	Providing access path to mounted storage (no valid file)
/opt/	Code for configuration interface services ('local_control/' directory)
/proc/	Providing access path to information related to active system and process
/root/	Home directory for root account Object files and shell scripts related to Bluetooth, Wi-Fi, Zigbee Basic configuration files ('config/hub.conf')
/sbin/	Command executables related to system control and configuration
/sys/	Providing access path to system parameter

/tmp/	System log ('all.log', etc.) Temporary files	
/usr/	Auxiliary hierarchy for read-only user data (executables, libraries, architecture-independent files, etc.)	
/var/	Variable files (guessed to be symbolic-linked to other directories, such as '/tmp/')	
/	Root directory Shell command history ('.ash_history')	

Although no significant information related to the incident was found in the file system, **it was possible to identify some basic information** about the hub as in the following table.

**Table 32 – Basic information of Wink Hub**

Path	Information	
/root/wifi/nvram.conf	MAC Address of Wi-Fi NIC	00:90:4C:C5:12:38
/database/bd_addr	Bluetooth Address	00:21:CC:09:B7:C9
/tmp/all.log	Leased IP Address	10.20.30.22
/database/zigbee-version.txt	Zigbee Version	5.1.2.1
/database/authorized_keys	Authorized SSH Key	Key of 'fluffy@hogwarts'
/database/wpa_supplicant.conf	Saved Wi-Fi AP	SSID "ESC-IoT" (PSK="esc_iot_2018")

### 3.1.5 Amazon Echo

**A smart speaker called 'Amazon Echo' is placed at the entrance of the Drug Lab.** This speaker responds to the name 'Alexa' (wake word) and executes the user's voice command. Through the speaker, it is possible to not only check the weather or listen to music, but also control various IoT devices.

On this device, data was extracted through a tool called 'Cloud-based IoT Forensic Toolkit (CIIFT)'. The tool collects data from both cloud side and client side using the Amazon smart speaker's API, which allows you to view user information, devices associated with speakers, Wi-Fi information, voice command history, and more.

The output files made by CIIFT are listed in the following table.

**Table 33 – Description of CIIFT output files**

Filename	Description
Evidence_Library/AmazonAlexaCloud/	Raw data (JSON file and command voice file)
cift_amazon_alexa.db	Database with extracted results
cift_amazon_alexa_ACCOUNT.csv	Account information
cift_amazon_alexa_ACQUIRED_FILE.csv	Source data acquisition method, API path, file description, etc.
cift_amazon_alexa_ALEXA_DEVICE.csv	Basic device information
cift_amazon_alexa_SETTING_MISC.csv	Miscellaneous device settings
cift_amazon_alexa_SETTING_WIFI.csv	Wi-Fi connection settings
cift_amazon_alexa_SKILL.csv	Linked device control (skill) information
cift_amazon_alexa_TIMELINE.csv	Usage history such as voice command
last_progress_log.txt	CIIFT extraction progress log

### **3.1.5.1 Basic Information**

- Accounts
  - Username: Jessie Pinkman, E-mail: jpinkman2018@gmail.com, ID: A2F07N8TDIAK5U
- Wi-Fi Connection
  - SSID: 'ESC-IoT', Security Method: WPA-PSK, PSK: 'esc\_iot\_2018'
- Linked Device Control (Skill)
  - Arlo, Nest Camera: Live streaming of camera image by Echo Show, Echo Spot, Fire TV or Fire tablets
  - Reuters TV (U.S.): Up-to-date news from Reuters TV.
  - Weather: A brief update on today's weather conditions
  - Wink: Controlling Wink-compatible lights, outlets, switches, locks, and thermostats
  - iSmartAlarm: Arming or disarming alarm system

### **3.1.5.2 Activity Timeline**

The 'cift\_amazon\_alexa\_TIMELINE.csv' file contains history of the voice commands and responses in chronological order.

The first time history available on the timeline is March 25, 2018 at 8:45:07 pm.

The following table summarizes the history of voice commands and Alexa's answers (wake words are ignored).

**Table 34 – History of voice commands and Alexa's answers**

Date	Type	Contents
2018-05-15 11:27:08	User's command	alexa tell i smart alarm to arm my system
2018-05-15 11:27:09	Alexa's answer	your home info error please relink.
2018-05-15 11:28:57	User's command	tell i. smart alarm to arm my system
2018-05-15 11:28:58	Alexa's answer	Your Door is open, Are you sure you want to arm your system?
2018-05-15 11:29:04	User's command	no
2018-05-15 11:29:05	Alexa's answer	Arming canceled.
2018-05-15 15:05:50	User's command	what time is it
2018-05-15 15:05:51	Alexa's answer	The time is 3:05 PM.
2018-05-15 15:06:01	User's command	alexa say hello
2018-05-15 15:06:01	Alexa's answer	Hi!
2018-05-15 15:06:06	User's command	alexa how are you
2018-05-15 15:06:06	Alexa's answer	All systems go!
2018-05-15 15:06:15	User's command	what's the weather for today
2018-05-15 15:06:15	Alexa's answer	Right now in Le Pigeon, it's 16 degrees Celsius with mostly cloudy skies. Today's forecast has lots of clouds and thunderstorms, with a high of 17 degrees and a low of 9 degrees.
2018-05-15 15:06:49	User's command	what's up
2018-05-15 15:06:51	Alexa's answer	Good afternoon,
2018-05-15 15:06:51	Alexa's answer	It's 3:06 PM.
2018-05-15 15:06:51	Alexa's answer	Here's what's up.

Date	Type	Contents
2018-05-15 15:06:51	Alexa's answer	By the way, I just read an interesting story. Canadian actress Margot Kidder, famous for playing Lois Lane, is dead at 69. Do you want to hear more about that?
2018-05-15 15:07:08	User's command	yes
2018-05-15 15:07:09	Alexa's answer	Canadian actress Margot Kidder, famous for playing Lois Lane alongside Christopher Reeve's Superman, has passed away. She was 69.
2018-05-15 15:07:31	User's command	did you have another story
2018-05-15 15:07:31	Alexa's answer	Sorry, I don't know that one.
2018-05-15 15:07:43	User's command	well i see you soon
2018-05-15 15:07:44	Alexa's answer	-
2018-05-15 15:07:50	User's command	bye
2018-05-15 15:07:51	Alexa's answer	Goodbye.
2018-05-16 15:34:19	User's command	what's up
2018-05-16 15:34:19	Alexa's answer	Good afternoon,
2018-05-16 15:34:19	Alexa's answer	It's 3:34 PM.
2018-05-16 15:34:19	Alexa's answer	Here's what's up.
2018-05-16 15:34:19	Alexa's answer	By the way, I just read an interesting story. the cost of chicken wings has slumped 37% since fall, and now sits at a seasonal 4-year low. Do you want to hear more about that?
2018-05-16 15:34:38	User's command	yes
2018-05-16 15:34:39	Alexa's answer	Wholesale chicken wing prices in the U.S. reached as high as \$2 a pound in September, the highest ever in U.S. Department of Agriculture data for the past 7 years. Since then, the cost has slumped about 37% and sits at a seasonal 4-year low. The popularity of wings tends to peak during Super Bowl season.
2018-05-16 15:35:01	User's command	incredible
2018-05-16 15:35:02	Alexa's answer	I'm not quite sure how to help you with that.
2018-05-16 15:35:20	User's command	what's friendsgiving
2018-05-16 15:35:22	Alexa's answer	'friendsgiving' is usually defined as a thanksgiving feast held by and for someone's friends.
2018-05-16 15:36:06	User's command	what's four and six
2018-05-16 15:36:07	Alexa's answer	4 plus 6 equals 10.
2018-05-17 10:16:09	User's command	alexa play led zeppelin
2018-05-17 10:16:09	Alexa's answer	To play Spotify, link your premium account first using the Alexa App.
2018-05-17 10:16:20	User's command	Unknown (bustling sound)
2018-05-17 10:22:12	User's command	tell i. smart alarm to arm my system
2018-05-17 10:22:14	Alexa's answer	Your Door is open, Are you sure you want to arm your system?
2018-05-17 10:22:19	User's command	yes
2018-05-17 10:22:21	Alexa's answer	Your system will set to Arm in 30 seconds.

There are many voice commands irrelevant to the incident, but **one thing to keep in mind in the history is that someone tried to arm iSmartAlarm on the day the raid happened.**

At May 15, the speaker was started to be used in earnest, and ordered to arm iSmartAlarm for the first time.

**Here are records related to the iSmartAlarm arm command that we can see in the timeline:**

- ① 2018-05-15 11:27:08 Arm Command  
Echo informed that home info error occurred and requested relink to the alarm system.

- ② 2018-05-15 11:28:57 Arm Command  
 Echo asked: "Your Door is open, Are you sure you want to arm your system?"  
 User's Command : "No"  
 Arming cancelled.

- ③ **2018-05-17 10:22:12 Arm Command**  
 Echo asked: "Your Door is open, Are you sure you want to arm your system?"  
 User's Command : "**Yes**"

2018-05-17 10:22:21

Echo informed that the alarm system will be armed in 30 seconds.

Of three commands, it is only the last case that actually armed the alarm system.

The time the command occurred is very close to the time the raid took place, so we have to take account of other evidences to identify what happened near the time the arm command executed.

## 3.2 Network Level Analysis

This section explains the key information found through analysis of the packet file ('dfrws\_police.pcap') collected from the network at the crime scene.

For 57 seconds, from 15:36:25 to 15:37:23 on May 17, 2018, 4,249 packets (1,643,724 bytes) were occurred at the network of the crime scene after the incident.

Prior to the main analysis, in order to check the discrepancy between the system time of the host that collected the packet and the actual time, we examined packet 2011 which is one of the recorded NTP server packets. As shown in the figure below, the NTP server 72.21.192.213 recorded the transmit timestamp as 13:36:53.239973539 UTC, and the collected packet were recorded as 13:36:53.276234 UTC. It can be seen that the host was approximately 36 milliseconds faster than the NTP server. This is a negligible difference, so the analysis is performed without any time shift.

2005 2018-05-17 13:36:53.176373 10.20.30.23	72.21.192.213	NTP	90 NTP Version 4, client
2011 2018-05-17 13:36:53.276234 72.21.192.213	10.20.30.23	NTP	90 NTP Version 4, server
2675 2018-05-17 13:37:01.974980 10.20.30.15	217.147.208.1	NTP	90 NTP Version 3, client
2676 2018-05-17 13:37:01.979605 217.147.208.1	10.20.30.15	NTP	90 NTP Version 3, server
3741 2018-05-17 13:37:16.314752 10.20.30.23	66.135.44.92	NTP	90 NTP Version 4, client
3751 2018-05-17 13:37:16.470366 66.135.44.92	10.20.30.23	NTP	90 NTP Version 4, server

▼ Network Time Protocol (NTP Version 4, server)
> Flags: 0x24, Leap Indicator: no warning, Version number: NTP Version 4, Mode: server
<a href="#">[Request In: 2005]</a>
[Delta Time: 0.099861000 seconds]
Peer Clock Stratum: secondary reference (3)
Peer Polling Interval: invalid (0)
Peer Clock Precision: 0.000001 seconds
Root Delay: 0.001953 seconds
Root Dispersion: 0.048447 seconds
Reference ID: 72.21.199.82
Reference Timestamp: May 17, 2018 13:33:59.840891842 UTC
Origin Timestamp: Apr 4, 2018 10:06:15.959881642 UTC
Receive Timestamp: May 17, 2018 13:36:53.238101989 UTC
Transmit Timestamp: May 17, 2018 13:36:53.239973539 UTC

Figure 38 – NTP server packet 2011 in the packet file

### 3.2.1 Entities

There are 9 devices in the network that appear as sender or receiver, distinguished by MAC addresses.

The following table summarizes IP address and manufacturer, corresponding to each MAC address.

Manufacturers are can be identified by the Organization Unique Identifier (OUI), which is the top 24 bits of the MAC Address.

**Table 35 – Entities identified by MAC address**

MAC Address	Packets	Bytes	IP Address	Vendor Identified by OUI
B8:27:EB:0E:3B:45	4237	1642072	Various (10.20.30.1)	Raspberry Pi Foundation
18:B4:30:61:C9:EF	3924	1593230	10.20.30.13	Nest Labs Inc.
AC:5F:3E:73:E3:78	129	18357	10.20.30.21	SAMSUNG ELECTRO-MECHANICS(THAILAND)
D8:FB:5E:E1:01:92	115	14208	10.20.30.15	ASKEY COMPUTER CORP
74:75:48:96:23:24	34	4977	10.20.30.23	Amazon Technologies Inc.
01:00:5E:7F:FF:FA	20	10322	239.255.255.250	IPv4 Multicast
08:02:8E:FF:75:4F	20	2030	10.20.30.17	NETGEAR
B4:79:A7:25:02:FA	13	1368	10.20.30.22	SAMSUNG ELECTRO-MECHANICS(THAILAND)
18:B4:30:99:9F:85	6	884	10.20.30.19	Nest Labs Inc.

Based on the above results, the following inference can be made as to which MAC address corresponds to which device of the Drug Lab:

- **B8:27:EB:0E:3B:45 : Router**
  - Given that this address appeared in almost every packet and as multiple source or destination IP addresses, we can assume that the corresponding device acted as a router connected to the external network.
  - Considering that IP address band of other devices is 10.20.30.0/24, the IP address of this device in the internal network is 10.20.30.1.
  - IP addresses that appeared with this MAC address are:
    - ◆ As Source and Destination MAC Address:  
'31.13.64.16', '52.46.156.66', '34.224.5.65', '54.152.107.0', '130.223.8.20',  
'144.76.81.240', '31.13.64.35', '10.20.30.1', '217.147.208.1', '66.135.44.92',  
'172.217.16.142', '72.21.192.213', '172.217.23.106'
    - ◆ Only As Source MAC Address:  
'216.58.205.174', '54.72.123.194', '35.195.59.182'
    - ◆ Only As Destination MAC Address:  
'23.23.78.17', '23.23.189.37'
- **18:B4:30:61:C9:EF (10.20.30.13): Nest Cam**  
**18:B4:30:99:9F:85 (10.20.30.19): Nest Protect**
  - The manufacturer of OUI '18:B4:30' is 'Nest Labs Inc.'
  - We can identify MAC Address of the Nest Cam by the photograph taken by the Pinkman's smartphone.
- **D8:FB:5E:E1:01:92 (10.20.30.15): QBee Camera**
  - The manufacturer of OUI 'D8:FB:5E' is 'ASKEY COMPUTER CORP', where sells QBee Camera.

- We can identify MAC Address of the camera by the screenshot taken by the Pinkman's smartphone.
- **74:75:48:96:23:24 (10.20.30.23): Amazon Echo**
  - The manufacturer of OUI '74:75:48' is 'Amazon Technologies Inc.'
- **08:02:8E:FF:75:4F (10.20.30.17): Arlo Base Station**
  - The manufacturer of OUI '08:02:8E' is 'NETGEAR', where sells Arlo security camera.
  - We can identify MAC Address of the base station by the NVRAM settings acquired from it.
- **AC:5F:3E:73:E3:78 (10.20.30.21) : Jessie Pinkman's smartphone**  
**B4:79:A7:25:02:FA (10.20.30.22) : Wink Hub**
  - The manufacturer of OUI 'AC:5F:3E' and 'B4:79:A7' is 'SAMSUNG ELECTRO-MECHANICS(THAILAND)'.
  - We can identify MAC Address of the smartphone by the setting file stored in it.
  - We can identify MAC Address of the hub by the setting file stored in it.

### 3.2.2 Conversations

Conversations among identified entities are listed in the following table.

**Table 36 – Conversations among entities**

Address A	Address B	Packets	A → B		B → A	
			Packets	Bytes	Packets	Bytes
18:B4:30:61:C9:EF	B8:27:EB:0E:3B:45	3924	2132	1474834	1792	118396
AC:5F:3E:73:E3:78	B8:27:EB:0E:3B:45	117	56	6595	61	10110
D8:FB:5E:E1:01:92	B8:27:EB:0E:3B:45	103	52	5024	51	7532
74:75:48:96:23:24	B8:27:EB:0E:3B:45	34	26	4115	8	862
01:00:5E:7F:FF:FA	B8:27:EB:0E:3B:45	20	0	0	20	10322
08:02:8E:FF:75:4F	B8:27:EB:0E:3B:45	20	10	1166	10	864
B4:79:A7:25:02:FA	B8:27:EB:0E:3B:45	13	6	643	7	725
AC:5F:3E:73:E3:78	D8:FB:5E:E1:01:92	12	8	1236	4	416
18:B4:30:99:9F:85	B8:27:EB:0E:3B:45	6	3	446	3	438

Of the total 4,249 packets, 4,079 packets (96%) are TCP and 86 are UDP. Most of the TCP communication data were generated by TLS communication. UDP packets were generated by SSDP (Simple Service Discovery Protocol), NTP (Network Time Protocol), DHCP (Dynamic Host Configuration Protocol), and DNS (Domain Name System).

Among the conversations that can be observed in the packet file, **the notable points are as follows:**

- The conversation with the highest number of packets is the connection between 10.20.30.13 (18:B4:30:61:C9:EF) and 35.195.59.182.
  - 2,132 packets (1,474k bytes) sent from 10.20.30.13 to 35.195.59.182, and 1,792 packets (118k bytes) vice versa, so 3,924 packets (92% of the total) were transmitted.
  - This communication continued throughout the packet collection process, but we don't know the contents of the communication because it is encrypted with the TLS protocol.
  - However, **since 10.20.30.13 is presumed to be the Nest Cam, it is possible to guess**

that the image recorded by the camera is being streamed through this connection.

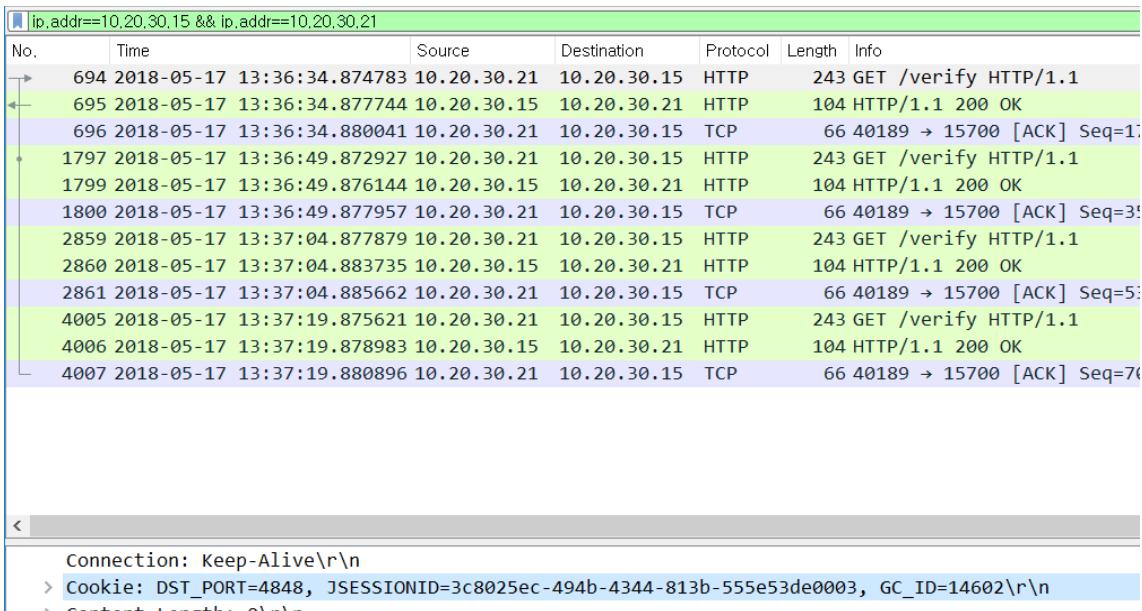


Figure 39 – Heartbeat packets sent to QBee camera (10.20.30.15)

- Total 12 packets were transmitted and received between 10.20.30.15 (D8:FB:5E:E1:01:92) and 10.20.30.21 (AC:5F:3E:73:E3:78) as shown in the above figure.
  - 10.20.30.15 is estimated as the QBee Camera, and 10.20.30.21 is estimated as the Pinkman's smartphone.
  - In this conversation, **10.20.30.21** sends an HTTP request to the '/verify' address at a fixed time interval, which is a **heartbeat packet** sent to the camera to receive data streamed from the QBee Camera. At this time, **the packet includes cookies** for authentication.
  - Since the cookies are transmitted through the unencrypted HTTP communication, they can be easily sniffed as shown in the lower part of the figure.
  - Regrettably, **a vulnerability (CVE-2018-16225) has been found that can reuse cookies to execute arbitrary commands on the camera** [25]. YouTube video 'QBee Network Communication Vulnerability' [26] shows that the camera can be disabled (switched to private mode) by using this vulnerability. **This implies that it is possible for the QBee Camera to be remotely disabled by the vulnerability or other means.**

### 3.3 Correlations of Findings

This section describes the spatial and temporal information of the crime scene that can be derived from a variety of the findings from device level analysis and network level analysis.

#### 3.3.1 Network Structure of the Drug Lab

By combining device information and device association information stored in various configuration files, photo files and databases found in device level analysis, and entities and conversations identified in network level analysis, it is possible to grasp how the physical (local network connection) and logical (device association) connections among the Drug Lab's devices are established.

The following tables summarize devices found in the Drug Lab and connections among the devices.

**Table 37 – Identified devices found in the Drug Lab**

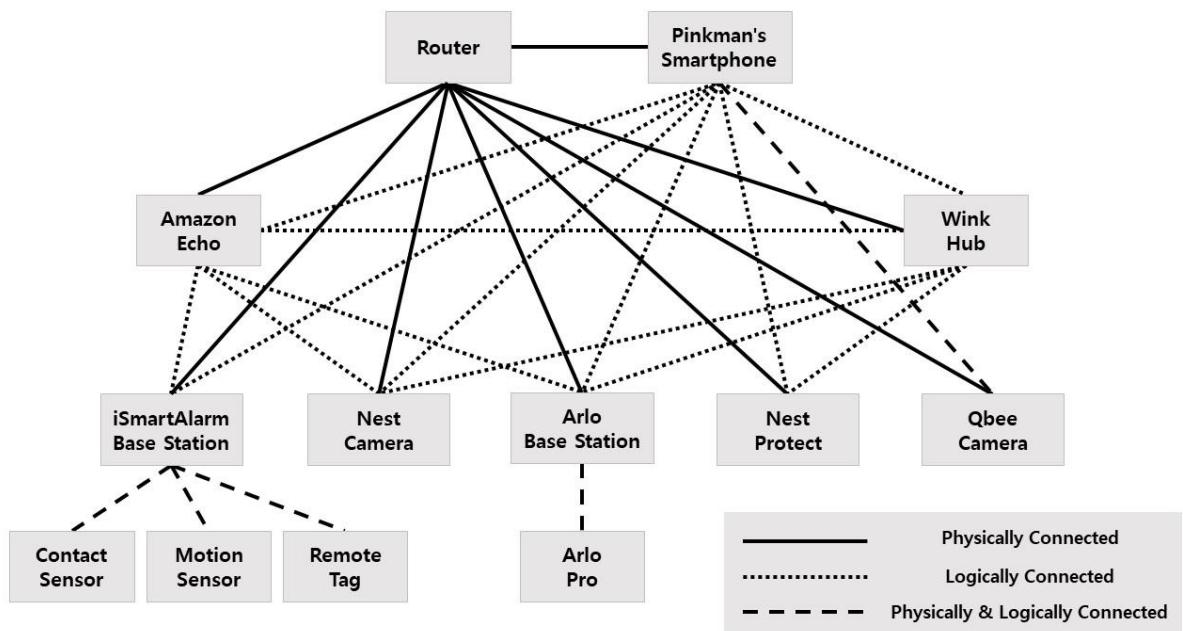
Device	Network Address (MAC [ / IP])		References
Router	B8:27:EB:0E:3B:45 / 10.20.30.1		3.2.1 Entities
Pinkman's Smartphone (SM-G925F)	AC:5F:3E:73:E3:78 / 10.20.30.21		3.1.1.1 Basic Information 3.2.1 Entities
iSmartAlarm Base Station (iPU3G)	00:4D:32:09:D9:E4		3.1.1.2 Media File
iSmartAlarm Contact Sensor	-		3.1.1.2 Media File
iSmartAlarm Motion Sensor (PIR3G)	-		3.1.1.2 Media File
iSmartAlarm Remote Tag	-		3.1.1.2 Media File
Arlo Base Station (VMB4000)	in Router's Network	08:02:8E:FF:75:4F / 10.20.30.17	3.1.3.1 NVRAM Setting 3.1.3.3 NAND TAR Archive 3.2.1 Entities
	Itself as AP	08:02:8E:FF:75:4E	3.1.3.1 NVRAM Setting
Arlo Pro (VMC4030)	08:02:8E:FD:BD:CD		3.1.3.3 NAND TAR Archive
Wink Hub (PWHUB-WH01)	in Router's Network	B4:79:A7:25:02:FA / 10.20.30.22	3.1.1.2 Media File 3.2.1 Entities
	Itself as AP	00:90:4C:C5:12:38	3.1.4 Wink Hub
Amazon Echo	74:75:48:96:23:24 / 10.20.30.23		3.2.1 Entities
Nest Cam (A0005)	18:B4:30:61:C9:EF / 10.20.30.13		3.1.1.2 Media File 3.2.1 Entities
Nest Protect (A14)	18:B4:30:99:9F:85 / 10.20.30.19		3.2.1 Entities
QBee Camera	D8:FB:5E:E1:01:92 / 10.20.30.15		3.1.1.2 Media File 3.2.1 Entities

**Table 38 – Identified physical (local network connection) and logical (device association) connections among the Drug Lab's devices**

Type	Connection	Type	References
Physical	Router (Internet)	Wi-Fi / Ethernet	3.2.2 Conversations
	↔ Nest Cam		3.1.1.1 Basic Information
	↔ Pinkman's Smartphone		3.2.2 Conversations
	↔ QBee Camera		3.1.1.2 Media File
	↔ Amazon Echo		3.2.2 Conversations
			3.1.5.1 Basic Information

			3.2.2 Conversations
	↔ Nest Protect		3.2.2 Conversations
	↔ Arlo Base Station		3.1.3.1 NVRAM Setting 3.2.2 Conversations
	↔ Wink Hub		3.2.2 Conversations
	(↔ iSmartAlarm Base Station)		Specification
Physical & Logical	Pinkman's Smartphone		
	↔ QBee Camera	Wi-Fi + Application	3.1.1.2 Media File 3.2.2 Conversations
	iSmartAlarm Base Station		
	↔ Contact Sensor	Proprietary RF [27] + Association	3.1.1.2 Media File Specification
	↔ Motion Sensor		
	↔ Remote Tag		
	Arlo Base Station (AP)		
	↔ Arlo Pro	Wi-Fi + Association	Specification
Logical	Pinkman's Smartphone		
	↔ iSmartAlarm	Application	3.1.1.3 Application Data
	↔ Arlo		
	↔ Wink Hub		
	↔ Amazon Echo		
	↔ Nest Protect & Cam		
	Amazon Echo		
	↔ iSmartAlarm	Alexa Skills	3.1.5.1 Basic Information
	↔ Nest Cam		
	↔ Arlo Pro		
	↔ Wink Hub		
	Wink Hub		
	↔ Nest Cam	Association	3.1.1.3 Application Data
	↔ Nest Protect		
	↔ Arlo Pro		

Based on the knowledge about the devices and the connections identified, **we illustrate a diagram that shows the network structure of the Drug Lab.**



**Figure 40 – Diagram of the network structure of the Drug Lab**

### 3.3.2 Detailed Timeline

In Section 2.1, we showed the brief timeline of major events happened in the day of invasion. **At this part, we organize our findings chronologically with more details.** This timeline includes not only the main actions in the scenario, but also the creation or registration time of a particular data, the time the sensor has detected a particular action, and so on.

**Table 39 – Detailed timeline of the scenario**

No	Date	Event	Reference
1	2018-03-14 15:31:38	key 'dateCreated' of Arlo Pro at Arlo App	3.1.1.3 - Arlo
2	2018-03-16 12:08:45	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
3	2018-03-16 14:24:18	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
4	2018-04-09 09:38:23	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
5	2018-04-27 13:12:16	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
6	2018-04-27 15:59:06	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
7	2018-04-29 10:50:51	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
8	2018-05-02 07:36:45	Nest Protect : protect_power_out_now	3.1.1.3 - Nest
9	2018-05-14 11:33:07	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
10	2018-05-14 11:44:31	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
11	2018-05-15 10:53:39	Nest Protect : Smoke and CO not detected	3.1.1.3 - Wink - Smart Home
12	2018-05-15 10:53:39	Nest Protect is registered at Wink Hub	3.1.1.3 - Wink - Smart Home
13	2018-05-15 10:53:40	Nest Cam : Capturing video on	3.1.1.3 - Wink - Smart Home
14	2018-05-15 10:53:40	Nest Cam is registered at Wink Hub	3.1.1.3 - Wink - Smart Home
15	2018-05-15 11:10:25	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
16	2018-05-15 11:12:15	Nest Cam : Capturing video off	3.1.1.3 - Wink - Smart Home

No	Date	Event	Reference
17	2018-05-15 11:14:36	key 'dateCreated' of Arlo Base Station at Arlo App	3.1.1.3 - Arlo
18	2018-05-15 11:17:29	Nest Cam : Capturing video on	3.1.1.3 - Wink - Smart Home
19	2018-05-15 11:27:08	User's command : alexa tell i smart alarm to arm my system	3.1.5.2 Activity Timeline
20	2018-05-15 11:27:09	Alexa's answer : your home info error please relink.	3.1.5.2 Activity Timeline
21	2018-05-15 11:28:57	User's command : tell i. smart alarm to arm my system	3.1.5.2 Activity Timeline
22	2018-05-15 11:28:58	Alexa's answer : Your Door is open, Are you sure you want to arm your system?	3.1.5.2 Activity Timeline
23	2018-05-15 11:29:04	User's command : no	3.1.5.2 Activity Timeline
24	2018-05-15 11:29:05	Alexa's answer : Arming canceled.	3.1.5.2 Activity Timeline
25	2018-05-15 11:33:39	Nest Cam : Capturing video off	3.1.1.3 - Wink - Smart Home
26	2018-05-15 11:38:56	Nest Cam : Capturing video on and motion detected	3.1.1.3 - Wink - Smart Home
27	2018-05-15 11:39:03	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
28	2018-05-15 14:38:20	JPinkman set alarm mode to ARM	3.1.1.3 - iSmartAlarm
29	2018-05-15 14:41:10	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
30	2018-05-15 14:41:28	Contact sensor : Door Open (Alarm)	3.1.2.1 Diagnostic Logs
31	2018-05-15 14:41:35	JPinkman set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
32	2018-05-15 14:51:08	[Nest Cam (frame 6 of video 78)] It seems that someone is coming in this room.	3.1.1.3 - Nest
33	2018-05-15 14:51:18	[Nest Cam (frame 7 of video 78)] A man A is going out.	3.1.1.3 - Nest
34	2018-05-15 14:51:21	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
35	2018-05-15 14:58:55	[Nest Cam (frame 12 of video 78)] Another man B is coming in the room.	3.1.1.3 - Nest
36	2018-05-15 14:59:11	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
37	2018-05-15 14:59:51	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
38	2018-05-15 15:00:38	[Nest Cam (frame 17 of video 78)] the man B is going out.	3.1.1.3 - Nest
39	2018-05-15 15:00:54	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
40	2018-05-15 15:03:01	[Nest Cam (frame 18 of video 78)] The room is empty.	3.1.1.3 - Nest
41	2018-05-15 15:03:11	pandadodu set alarm mode to PANIC	3.1.1.3 - iSmartAlarm
42	2018-05-15 15:03:28	pandadodu set alarm mode to PANIC	3.1.1.3 - iSmartAlarm
43	2018-05-15 15:03:36	pandadodu set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
44	2018-05-15 15:03:43	pandadodu set alarm mode to PANIC	3.1.1.3 - iSmartAlarm
45	2018-05-15 15:03:47	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
46	2018-05-15 15:04:08	[Nest Cam (frame 19 of video 78)] The door opened slightly.	3.1.1.3 - Nest
47	2018-05-15 15:04:43	pandadodu set alarm mode to HOME	3.1.1.3 - iSmartAlarm
48	2018-05-15 15:04:57	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
49	2018-05-15 15:04:59	Contact sensor : Door Open (Alarm)	3.1.2.1 Diagnostic Logs
50	2018-05-15 15:05:27	JPinkman set alarm mode to DISARM	3.1.1.3 - iSmartAlarm

No	Date	Event	Reference
51	2018-05-15 15:05:27	[Nest Cam (frame 1 of video 79)] the man B appeared again.	3.1.1.3 - Nest
52	2018-05-15 15:05:41	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
53	2018-05-15 15:05:50	User's command : what time is it	3.1.5.2 Activity Timeline
54	2018-05-15 15:05:51	Alexa's answer : The time is 3:05 PM.	3.1.5.2 Activity Timeline
55	2018-05-15 15:06:01	User's command : alexa say hello	3.1.5.2 Activity Timeline
56	2018-05-15 15:06:01	Alexa's answer : Hi!	3.1.5.2 Activity Timeline
57	2018-05-15 15:06:06	User's command : alexa how are you	3.1.5.2 Activity Timeline
58	2018-05-15 15:06:06	Alexa's answer : All systems go!	3.1.5.2 Activity Timeline
59	2018-05-15 15:06:15	User's command : what's the weather for today	3.1.5.2 Activity Timeline
60	2018-05-15 15:06:15	Alexa's answer : Right now in Le Pigeon, it's 16 degrees Celsius with mostly cloudy skies. Today's forecast has lots of clouds and thunderstorms, with a high of 17 degrees and a low of 9 degrees.	3.1.5.2 Activity Timeline
61	2018-05-15 15:06:40	[Nest Cam (frame 2 of video 79)] The room is empty again with the door closed.	3.1.1.3 - Nest
62	2018-05-15 15:06:49	User's command : what's up	3.1.5.2 Activity Timeline
63	2018-05-15 15:06:51	Alexa's answer : Good afternoon,	3.1.5.2 Activity Timeline
64	2018-05-15 15:06:51	Alexa's answer : It's 3:06 PM.	3.1.5.2 Activity Timeline
65	2018-05-15 15:06:51	Alexa's answer : Here's what's up.	3.1.5.2 Activity Timeline
66	2018-05-15 15:06:51	Alexa's answer : By the way, I just read an interesting story. Canadian actress Margot Kidder, famous for playing Lois Lane, is dead at 69. Do you want to hear more about that?	3.1.5.2 Activity Timeline
67	2018-05-15 15:07:08	User's command : yes	3.1.5.2 Activity Timeline
68	2018-05-15 15:07:09	Alexa's answer : Canadian actress Margot Kidder, famous for playing Lois Lane alongside Christopher Reeve's Superman, has passed away. She was 69.	3.1.5.2 Activity Timeline
69	2018-05-15 15:07:31	User's command : did you have another story	3.1.5.2 Activity Timeline
70	2018-05-15 15:07:31	Alexa's answer : Sorry, I don't know that one.	3.1.5.2 Activity Timeline
71	2018-05-15 15:07:43	User's command : well i see you soon	3.1.5.2 Activity Timeline
72	2018-05-15 15:07:44	Alexa's answer : -	3.1.5.2 Activity Timeline
73	2018-05-15 15:07:50	User's command : bye	3.1.5.2 Activity Timeline
74	2018-05-15 15:07:51	Alexa's answer : Goodbye.	3.1.5.2 Activity Timeline
75	2018-05-15 15:09:06	[Nest Cam (frame 4 of video 79)] the man A appeared again.	3.1.1.3 - Nest
76	2018-05-15 15:10:20	[Nest Cam (frame 5 of video 79)] The room is empty again (3rd) with the door closed.	3.1.1.3 - Nest
77	2018-05-15 15:10:24	Creation time of the cached still images of Nest Cam (from Wink Hub App)	3.1.1.3 - Wink - Smart Home
78	2018-05-15 15:10:57	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
79	2018-05-15 15:11:05	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
80	2018-05-15 15:11:06	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
81	2018-05-15 15:11:06	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs

No	Date	Event	Reference
82	2018-05-15 15:16:00	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
83	2018-05-16 02:49:53	Arlo Pro is registered at Wink Hub	3.1.1.3 - Wink - Smart Home
84	2018-05-16 02:49:55	Arlo Pro : Armed	3.1.1.3 - Wink - Smart Home
85	2018-05-16 15:32:17	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
86	2018-05-16 15:32:46	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
87	2018-05-16 15:33:33	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
88	2018-05-16 15:33:55	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
89	2018-05-16 15:34:19	User's command : what's up	3.1.5.2 Activity Timeline
90	2018-05-16 15:34:19	Alexa's answer : Good afternoon,	3.1.5.2 Activity Timeline
91	2018-05-16 15:34:19	Alexa's answer : It's 3:34 PM.	3.1.5.2 Activity Timeline
92	2018-05-16 15:34:19	Alexa's answer : Here's what's up.	3.1.5.2 Activity Timeline
93	2018-05-16 15:34:19	Alexa's answer : By the way, I just read an interesting story. the cost of chicken wings has slumped 37% since fall, and now sits at a seasonal 4-year low. Do you want to hear more about that?	3.1.5.2 Activity Timeline
94	2018-05-16 15:34:38	User's command : yes	3.1.5.2 Activity Timeline
95	2018-05-16 15:34:39	Alexa's answer : Wholesale chicken wing prices in the U.S. reached as high as \$2 a pound in September, the highest ever in U.S. Department of Agriculture data for the past 7 years. Since then, the cost has slumped about 37% and sits at a seasonal 4-year low. The popularity of wings tends to peak during Super Bowl season.	3.1.5.2 Activity Timeline
96	2018-05-16 15:34:45	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
97	2018-05-16 15:34:49	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
98	2018-05-16 15:35:01	User's command : incredible	3.1.5.2 Activity Timeline
99	2018-05-16 15:35:02	Alexa's answer : I'm not quite sure how to help you with that.	3.1.5.2 Activity Timeline
100	2018-05-16 15:35:10	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
101	2018-05-16 15:35:20	User's command : what's friendsgiving	3.1.5.2 Activity Timeline
102	2018-05-16 15:35:22	Alexa's answer : 'friendsgiving' is usually defined as a thanksgiving feast held by and for someone's friends.	3.1.5.2 Activity Timeline
103	2018-05-16 15:35:24	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
104	2018-05-16 15:36:06	User's command : what's four and six	3.1.5.2 Activity Timeline
105	2018-05-16 15:36:07	Alexa's answer : 4 plus 6 equals 10.	3.1.5.2 Activity Timeline
106	2018-05-16 15:36:27	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
107	2018-05-16 15:37:35	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
108	2018-05-16 15:39:55	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
109	2018-05-16 15:47:14	TheBoss set alarm mode to PANIC	3.1.1.3 - iSmartAlarm
110	2018-05-16 15:47:15	TheBoss set alarm mode to PANIC	3.1.1.3 - iSmartAlarm
111	2018-05-16 15:47:17	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
112	2018-05-16 15:47:18	TheBoss set alarm mode to HOME	3.1.1.3 - iSmartAlarm

No	Date	Event	Reference
113	2018-05-16 15:47:19	Contact sensor : Door Open (Alarm)	3.1.2.1 Diagnostic Logs
114	2018-05-16 15:47:33	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
115	2018-05-16 15:47:40	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
116	2018-05-16 15:47:53	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
117	2018-05-16 15:49:54	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
118	2018-05-16 15:50:13	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
119	2018-05-16 15:52:10	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
120	2018-05-16 15:52:45	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
121	2018-05-16 15:53:10	JPinkman set alarm mode to HOME	3.1.1.3 - iSmartAlarm
122	2018-05-16 15:53:31	JPinkman set alarm mode to ARM	3.1.1.3 - iSmartAlarm
123	2018-05-16 15:55:07	JPinkman set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
124	2018-05-16 15:55:11	JPinkman set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
125	2018-05-16 15:55:17	JPinkman set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
126	2018-05-16 15:55:27	JPinkman set alarm mode to ARM	3.1.1.3 - iSmartAlarm
127	2018-05-17 09:40:23	Creation time of the cached still images of Arlo Pro (from Arlo App)	3.1.1.3 - Arlo
128	2018-05-17 09:40:37	Creation time of the cached still images of Arlo Pro (from Wink Hub App)	3.1.1.3 - Wink - Smart Home
129	2018-05-17 09:44:53	Contact sensor : Door Open (Alarm)	3.1.2.1 Diagnostic Logs
130	2018-05-17 09:45:22	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
131	2018-05-17 09:47:18	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
132	2018-05-17 09:47:50	JPinkman set alarm mode to ARM	3.1.1.3 - iSmartAlarm
133	2018-05-17 10:09:52	Contact sensor : Door Open (Alarm)	3.1.2.1 Diagnostic Logs
134	2018-05-17 10:09:55	Motion sensor : Motion Detected (Alarm)	3.1.2.1 Diagnostic Logs
135	2018-05-17 10:09:57	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
136	2018-05-17 10:10:11	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
137	2018-05-17 10:10:45	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
138	2018-05-17 10:14:00	[Nest Cam (frame 20 of video 201)] Someone pushes a box toward the outside of the room.	3.1.1.3 - Nest
139	2018-05-17 10:14:05	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
140	2018-05-17 10:14:12	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
141	2018-05-17 10:14:43	[Nest Cam (frame 1 of video 202)] People are coming in the room.	3.1.1.3 - Nest
142	2018-05-17 10:15:00	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
143	2018-05-17 10:15:21	[Nest Cam (frame 4 of video 202)] Many people gather in the room.	3.1.1.3 - Nest
144	2018-05-17 10:15:56	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
145	2018-05-17 10:15:58	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
146	2018-05-17 10:16:09	User's command : alexa play led zeppelin	3.1.5.2 Activity Timeline
147	2018-05-17 10:16:09	Alexa's answer : To play Spotify, link your premium account first using the	3.1.5.2 Activity Timeline

No	Date	Event	Reference
		Alexa App.	
148	2018-05-17 10:16:20	User's command : Unknown (bustling sound)	3.1.5.2 Activity Timeline
149	2018-05-17 10:16:41	[Nest Cam (frame 8 of video 202)] people are going outside.	3.1.1.3 - Nest
150	2018-05-17 10:16:59	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
151	2018-05-17 10:22:12	User's command : tell i. smart alarm to arm my system	3.1.5.2 Activity Timeline
152	2018-05-17 10:22:14	Alexa's answer : Your Door is open, Are you sure you want to arm your system?	3.1.5.2 Activity Timeline
153	2018-05-17 10:22:19	User's command : yes	3.1.5.2 Activity Timeline
154	2018-05-17 10:22:21	Alexa's answer : Your system will set to Arm in 30 seconds.	3.1.5.2 Activity Timeline
155	2018-05-17 10:22:22	JPinkman set alarm mode to ARM	3.1.1.3 - iSmartAlarm
156	2018-05-17 10:22:25	[Nest Cam (frame 9 of video 202)] The room is empty.	3.1.1.3 - Nest
157	2018-05-17 10:22:30	TheBoss set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
158	2018-05-17 10:22:36	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
159	2018-05-17 10:28:50	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
160	2018-05-17 10:30:20	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
161	2018-05-17 10:30:30	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
162	2018-05-17 10:30:36	[Nest Cam (frame 13 of video 202)] Two of them come in the room again.	3.1.1.3 - Nest
163	2018-05-17 10:30:50	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
164	2018-05-17 10:31:48	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
165	2018-05-17 10:32:50	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
166	2018-05-17 10:33:15	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
167	2018-05-17 10:33:37	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
168	2018-05-17 10:34:15	Contact sensor : Door Closed	3.1.2.1 Diagnostic Logs
169	2018-05-17 10:34:17	TheBoss set alarm mode to HOME	3.1.1.3 - iSmartAlarm
170	2018-05-17 10:34:31	pandadodu set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
171	2018-05-17 10:34:36	Contact sensor : Door Open	3.1.2.1 Diagnostic Logs
172	2018-05-17 10:34:39	[Nest Cam (frame 2 of video 203)] Another man come into the room.	3.1.1.3 - Nest
173	2018-05-17 10:34:44	Nest Cam : Motion detected	3.1.1.3 - Wink - Smart Home
174	2018-05-17 10:34:52	[Nest Cam (frame 3 of video 203)] The room is empty again.	3.1.1.3 - Nest
175	2018-05-17 10:35:32	Nest Cam : Loudness detected	3.1.1.3 - Wink - Smart Home
176	2018-05-17 10:35:53	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
177	2018-05-17 10:36:06	Nest Protect : Smoke detected	3.1.1.3 - Wink - Smart Home
178	2018-05-17 10:36:06	Nest Protect : protect_smoke_warn	3.1.1.3 - Nest

No	Date	Event	Reference
179	2018-05-17 10:36:20	Nest Protect : Smoke not detected	3.1.1.3 - Wink - Smart Home
180	2018-05-17 10:36:20	Nest Protect : protect_smoke_warn_clear	3.1.1.3 - Nest
181	2018-05-17 10:37:52	pandadodu set alarm mode to DISARM	3.1.1.3 - iSmartAlarm
182	2018-05-17 10:38:52	Arlo Pro : Motion detected	3.1.1.3 - Wink - Smart Home
183	2018-05-17 10:40:00	The police were alerted about the incident	1.1 - Challenge Scenario
184	2018-05-17 10:45:00	The forensic team is on scene	1.1 - Challenge Scenario

## 4 Developments

This chapter introduces newly developed digital forensic analysis tools designed by the need of the analysis process of the challenge.

We made parsers to extract valuable information from various artifacts of IoT devices appeared in the challenge, and some of parsed results can be integrated, organized, and visualized via user interface.

Our toolset consists of parsers, user interface engine for visualization, and environment for running tools, and is distributed by GitHub.

(GitHub repository address : <https://github.com/philgeun/TapiocaPearlo>)

**Details of the toolset composition are described below** (asterisk indicates that the tool is not developed by us):

- **Parsers ('/tools' directory)**
  - Wink Activity Parser: Retrieves the activity records in the database of 'Wink – Smart Home' application.
  - Nest Video Recovery Tool: Extracts video files or individual video frames that were captured by Nest Cam and saved in the database of 'Nest' application.
  - iSmartAlarm App Dairy Parser: Retrieves the 'dairy' records - system events such as arm, disarm, and panic (alarm) and sensor events - in the database of 'iSmartAlarm' application.
  - iSmartAlarm Base Station Server Stream Parser: Retrieves event records in the server stream of iSmartAlarm base station.
  - Amazon Alexa CIFT Tool Parser: Retrieves usage history in the database created by CIFT.
- **User Interface Engine**
  - \*Elasticsearch: Open-source, RESTful, distributed search and analytics engine built on Apache Lucene.
  - \*Kibana: Interface for visualizing Elasticsearch data and navigating the Elastic Stack.
- **Environment:** Packaged by \*Docker container

### 4.1 Initial Setup

Before downloading the toolset, you must install prerequisites as enumerated below:

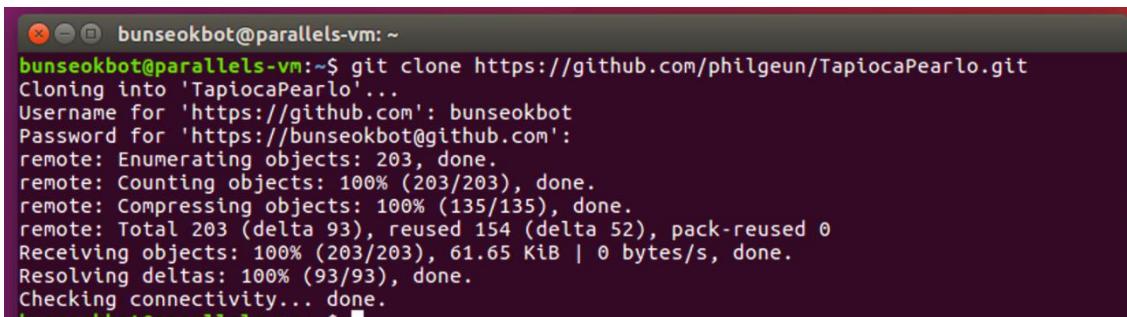
- Docker
- docker-compose

To use the toolset in Linux or MAC OS X, run the following command:  
(because the minimum count required for executing Elasticsearch is over 262144.)

```
sudo sysctl -w vm.max_map_count=262144
```

1. Clone the code from the GitHub repository

```
git clone https://github.com/philgeun/TapiocaPearlo.git
```

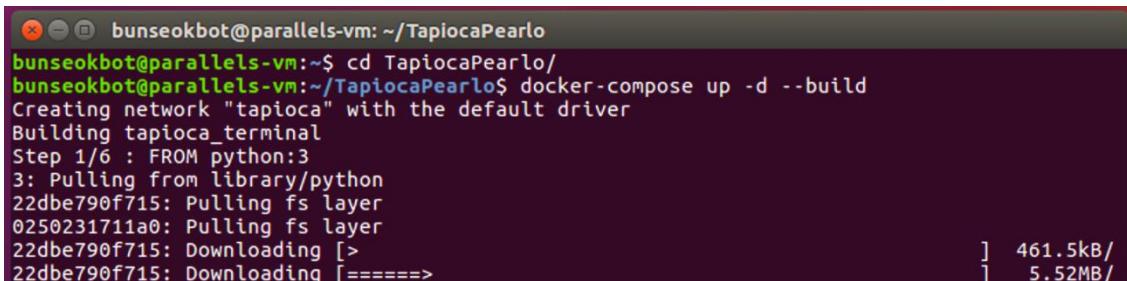


```
bunseokbot@parallels-vm:~$ git clone https://github.com/philgeun/TapiocaPearlo.git
Cloning into 'TapiocaPearlo'...
Username for 'https://github.com': bunseokbot
Password for 'https://bunseokbot@github.com':
remote: Enumerating objects: 203, done.
remote: Counting objects: 100% (203/203), done.
remote: Compressing objects: 100% (135/135), done.
remote: Total 203 (delta 93), reused 154 (delta 52), pack-reused 0
Receiving objects: 100% (203/203), 61.65 KiB | 0 bytes/s, done.
Resolving deltas: 100% (93/93), done.
Checking connectivity... done.
```

Figure 41 - Cloning the code from the GitHub repository

2. Activate all services using docker-compose

```
cd TapiocaPearlo/
docker-compose up -d --build
```

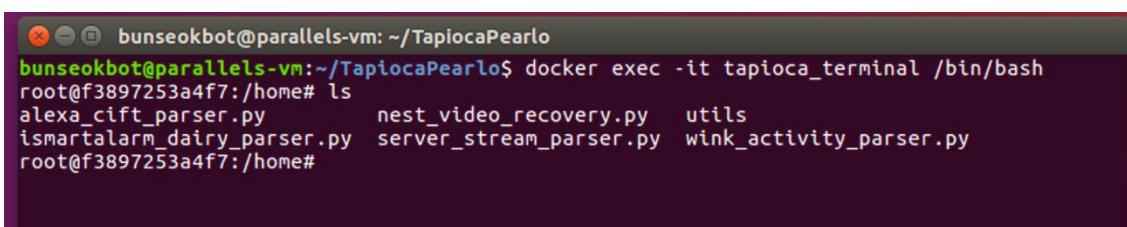


```
bunseokbot@parallels-vm:~/TapiocaPearlo$ cd TapiocaPearlo/
bunseokbot@parallels-vm:~/TapiocaPearlo$ docker-compose up -d --build
Creating network "tapioca" with the default driver
Building tapioca_terminal
Step 1/6 : FROM python:3
3: Pulling from library/python
22dbe790f715: Pulling fs layer
0250231711a0: Pulling fs layer
22dbe790f715: Downloading [>                                              ] 461.5kB/
22dbe790f715: Downloading [=====] 5.52MB/
```

Figure 42 - Activating all services using docker-compose

3. Connect to 'tapioca\_terminal' which provides the execution environment of the tool.  
Before you connect, you must move the artifacts you want to analyze into 'tools' directory.

```
docker exec -it tapioca_terminal /bin/bash
```



```
bunseokbot@parallels-vm:~/TapiocaPearlo$ docker exec -it tapioca_terminal /bin/bash
root@f3897253a4f7:/home# ls
alexa_cift_parser.py      nest_video_recovery.py    utils
ismartalarm_dairy_parser.py server_stream_parser.py  wink_activity_parser.py
root@f3897253a4f7:/home#
```

Figure 43 – Connecting to 'tapioca\_terminal'

## 4.2 Parser

### 1) Wink Activity Parser

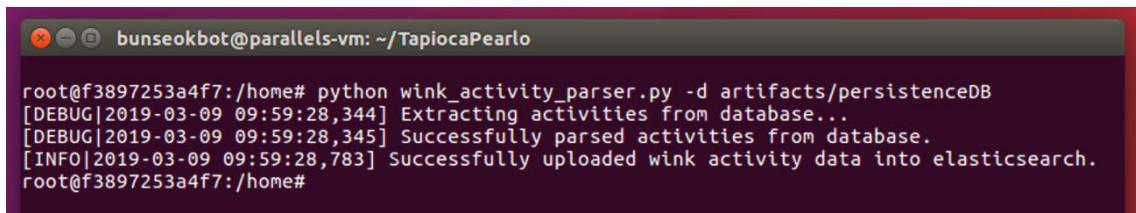
Wink Activity Parser ('wink\_activity\_parser.py') retrieves the activity records in the database of 'Wink – Smart Home' application and upload data into Elasticsearch.

- Usage

```
python wink_activity_parser.py -d <DB path>
```

- Arguments

- -d <DB path>: Path of 'persistenceDB' file extracted from 'Wink – Smart Home' application data.



```
bunseokbot@parallels-vm: ~/TapiocaPearl0
root@f3897253a4f7:/home# python wink_activity_parser.py -d artifacts/persistenceDB
[DEBUG|2019-03-09 09:59:28,344] Extracting activities from database...
[DEBUG|2019-03-09 09:59:28,345] Successfully parsed activities from database.
[INFO|2019-03-09 09:59:28,783] Successfully uploaded wink activity data into elasticsearch.
root@f3897253a4f7:/home#
```

Figure 44 – Running Wink Activity Parser

### 2) Nest Video Recovery Tool

Nest Video Recovery Tool ('nest\_video\_recovery.py') extracts video files or individual video frames that were captured by Nest Cam and saved in the database of 'Nest' application.

- Usage

```
python nest_video_recovery.py -d <DB path> [-o <output path>] [-m] [-a] [-f]
```

- Arguments

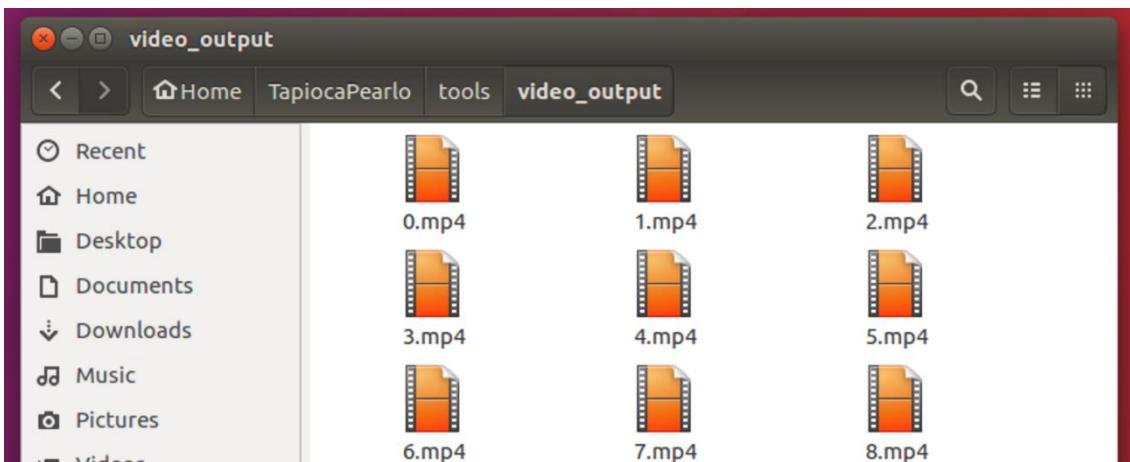
- -d <DB path>: Path of 'frame\_database' file extracted from 'Nest' application data.
- -o <output path> [optional]: Path where output files will be created (default: 'output')
- -m [optional]: Merge fragments of videos into one consequent video file.
- -a [optional]: Upload data into Elasticsearch.
- -f [optional]: Save each video frame individually with timestamp.

```

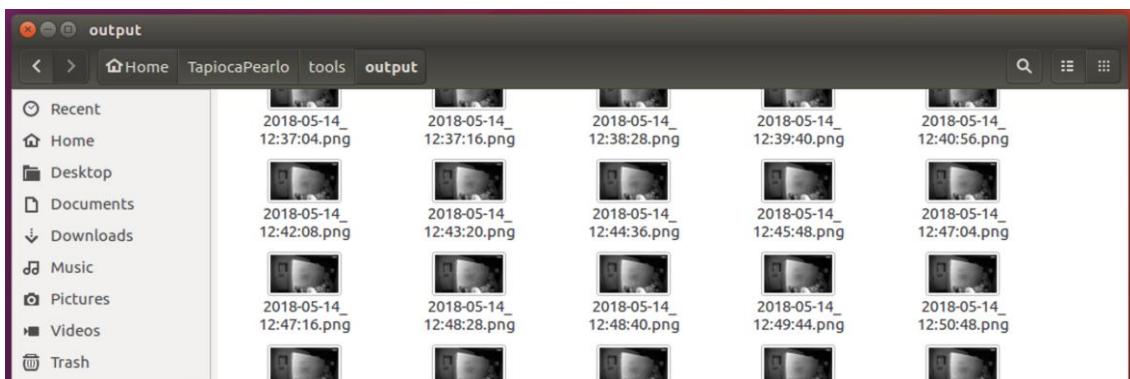
root@f3897253a4f7:/home# python nest_video_recovery.py -d artifacts/frame_database -o video_output
[DEBUG|2019-03-09 10:00:29,074] Extracting videos from database...
[INFO|2019-03-09 10:00:29,100] Successfully extracted 228 video files.
[INFO|2019-03-09 10:00:29,100] Converting video file codec format...
ffmpeg version 3.2.12-1~deb9u1 Copyright (c) 2000-2018 the FFmpeg developers
  built with gcc 6.3.0 (Debian 6.3.0-18+deb9u1) 20170516
  configuration: --prefix=/usr --extra-version='1~deb9u1' --toolchain=hardened --libdir=/usr/lib/x86_64-linux-gnu --incdir=/usr/include/x86_64-linux-gnu --enable-gpl --disable-stripping --enable-avresample --enable-avisynth --enable-gnutls --enable-ladspa --enable-libass --enable-libbluray --enable-libbs2b --enable-libcaca --enable-libcdio --enable-libebur128 --enable-libflite --enable-libfontconfig --enable-libfreetype --enable-libfribidi --enable-libgme --enable-libgsm --enable-libmp3lame --enable-libopenjpeg --enable-libopenmpt --enable-libopus --enable-libpulse --enable-librubberband --enable-libshine --enable-libsnappy --enable-libsoxr --enable-libspeex --enable-libssh --enable-libtheora --enable-libtwolame --enable-libvorbis --enable-libvpx --enable-libwavpack --enable-libwebp --enable-libx265 --enable-libxvid --enable-libzmq --enable-libzvbi --enable-omx --enable-openal --enable-opengl --enable-sdl2 --enable-libdc1394 --enable-libiec61883 --enable-chromaprint --enable-frei0r --enable-libopencv --enable-libx264 --enable-shared
    libavutil      55. 34.101 / 55. 34.101
    libavcodec     57. 64.101 / 57. 64.101
    libavformat    57. 56.101 / 57. 56.101
    libavdevice    57.  1.100 / 57.  1.100
    libavfilter     6. 65.100 / 6. 65.100

```

**Figure 45 – Running Nest Video Recovery Tool**



**Figure 46 – Output of Nest Video Recovery Tool**



**Figure 47 - Output of Nest Video Recovery Tool with frame-wise save argument**



**Figure 48 – Recovered video frame by Nest Video Recovery Tool**

### 3) iSmartAlarm App Dairy Parser

iSmartAlarm App Dairy Parser ('ismartalarm\_dairy\_parser.py') retrieves the 'Dairy' records - system events such as arm, disarm, and panic (alarm) and sensor events - in the database of 'iSmartAlarm' application and upload data into Elasticsearch.

- Usage

```
python ismartalarm_dairy_parser.py -d <DB path>
```

- Arguments

- -d <DB path>: Path of 'iSmartAlarm.DB' file extracted from 'iSmartAlarm' application data.

A terminal window with a dark background and light-colored text. The title bar says "bunseokbot@parallels-vm: ~/TapiocaPearl". The command "python ismartalarm\_dairy\_parser.py -d artifacts/iSmartAlarm.DB" is run. The output shows the script extracting diaries from the database, parsing the data, and successfully uploading IPUDairy and SensorDairy data to Elasticsearch. The timestamp in the logs indicates March 9, 2019, at 10:03:26.

**Figure 49 – Running iSmartAlarm App Dairy Parser**

### 4) iSmartAlarm Base Station Server Stream Parser

iSmartAlarm Base Station Server Stream Parser ('server\_stream\_parser.py') retrieves event records in the server stream of iSmartAlarm base station and upload data into Elasticsearch.

- Usage

```
python server_stream_parser.py -d <DB path>
```

- Arguments

- -d <DB path>: Path of 'server\_stream' file extracted from iSmartAlarm base station.

```

bunseokbot@parallels-vm: ~/TapiocaPearl
root@f3897253a4f7:/home# python server_stream_parser.py -s artifacts/server_stream
[INFO|2019-03-09 10:03:51,630] Start parsing iSmartAlarm diagnostics stream...
[INFO|2019-03-09 10:03:52,855] Successfully upload server_stream data.
root@f3897253a4f7:/home#

```

**Figure 50 – Running iSmartAlarm Base Station Server Stream Parser**

## 5) Amazon Alexa CIFT Tool Parser

Amazon Alexa CIFT Tool Parser ('alexa\_cift\_parser.py') retrieves usage history in the database created by CIFT and upload data into Elasticsearch.

- Usage

```
python alexa_cift_parser.py -d <DB path> -t <timezone='UTC+0200'>
```

- Arguments

- -d <DB path>: Path of 'iSmartAlarm.DB' file extracted from 'iSmartAlarm' application data.
- -t <timezone>: Timezone setting (must be 'UTC+0200' to get correct result)

```

bunseokbot@parallels-vm: ~/TapiocaPearl
root@f3897253a4f7:/home# python alexa_cift_parser.py -d artifacts/cift_amazon_alexa.db -t UTC+0200
[DEBUG|2019-03-09 10:04:24,978] Extracting activities from database...
[INFO|2019-03-09 10:04:24,989] Successfully parsed data from database.
[INFO|2019-03-09 10:04:25,164] Successfully uploaded data into elasticsearch.
root@f3897253a4f7:/home#

```

**Figure 51 – Running Amazon Alexa CIFT Tool Parser**

## 4.3 Visualization

You can use the user interface for visualization, provided by Elasticsearch and Kibana.

1. Connect to 'localhost:5601' with web browser.

**Add Data to Kibana**

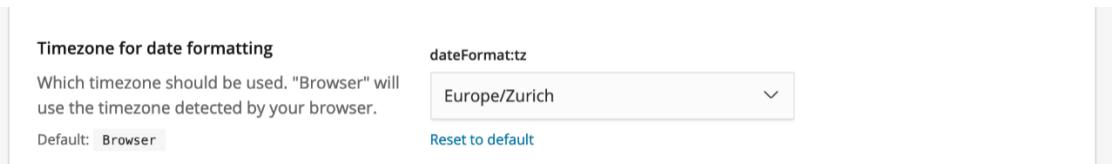
Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

 <b>APM</b> APM automatically collects in-depth performance metrics and errors from inside your applications.	 <b>Logging</b> Ingest logs from popular data sources and easily visualize in preconfigured dashboards.	 <b>Metrics</b> Collect metrics from the operating system and services running on your servers.	 <b>Security analytics</b> Centralize security events for interactive investigation in ready-to-go visualizations.
<a href="#">Add APM</a>	<a href="#">Add log data</a>	<a href="#">Add metric data</a>	<a href="#">Add security events</a>

**Figure 52 – Kibana main page**

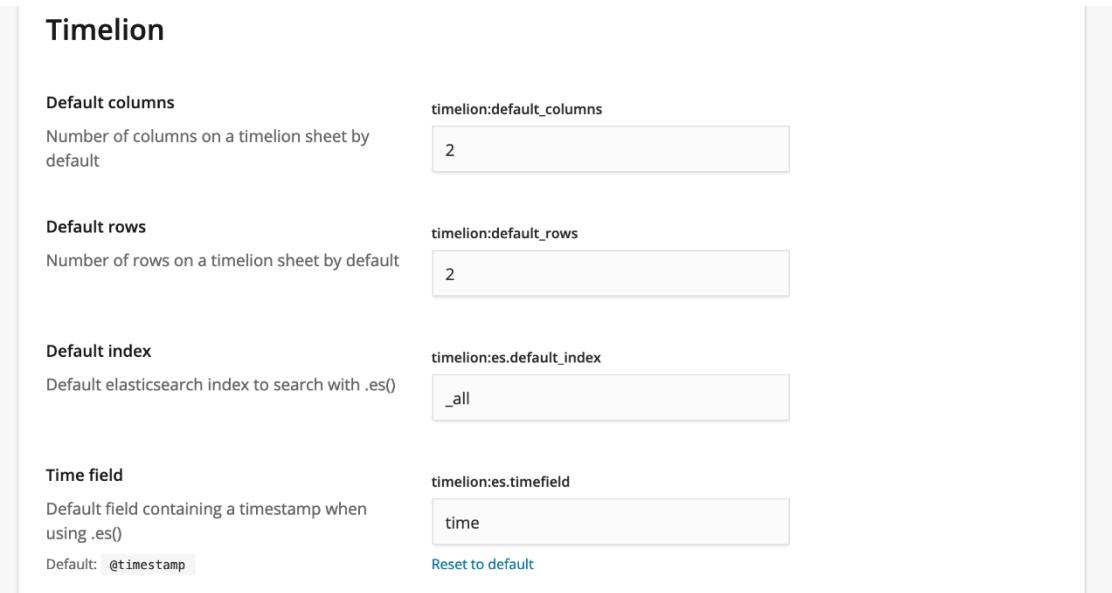
## 2. Configure basic settings

- i. Timezone: set correct timezone in 'Management -> Advanced Settings -> Timezone for date formatting'



**Figure 53 – Configure timezone for date formatting**

- ii. Timelion time field adjustment: set 'Management -> Advanced Settings -> Timelion -> Time field' to 'time'.



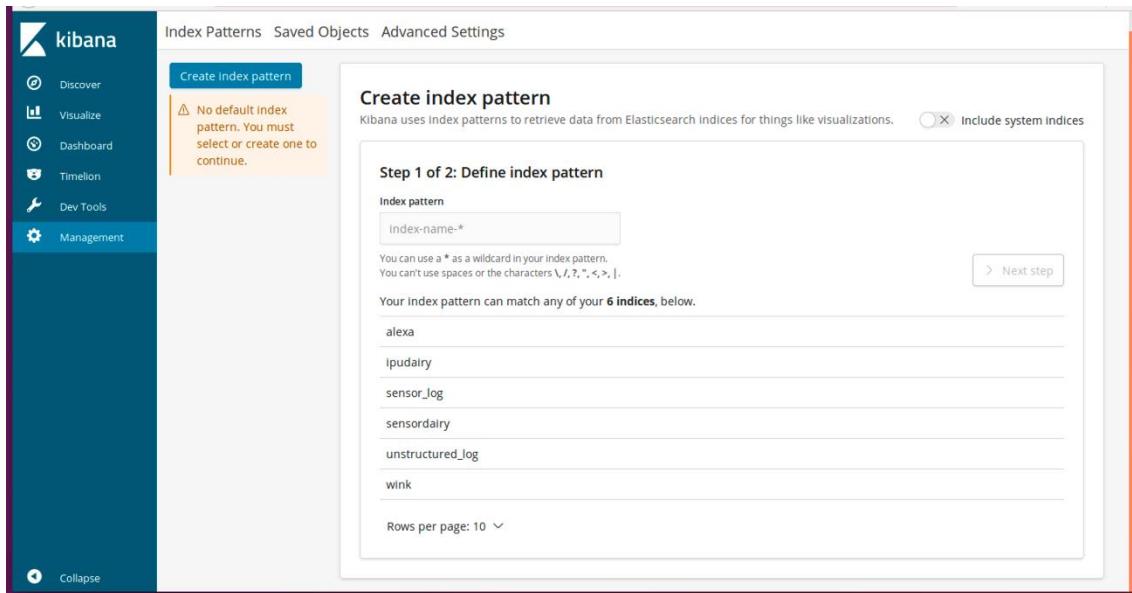
**Figure 54 – Timelion configuration**

## 3. Create index pattern saved in Elasticsearch.

You need to create a pattern of the index before you try to visualize data.

In 'Management -> Index Patterns', you could see the index of the information already stored in Elasticsearch.

Create the index pattern by inserting the indices as you can see in the below figure. ('alexa', 'ipudairy', 'sensor\_log', 'unstructured\_log', 'wink')

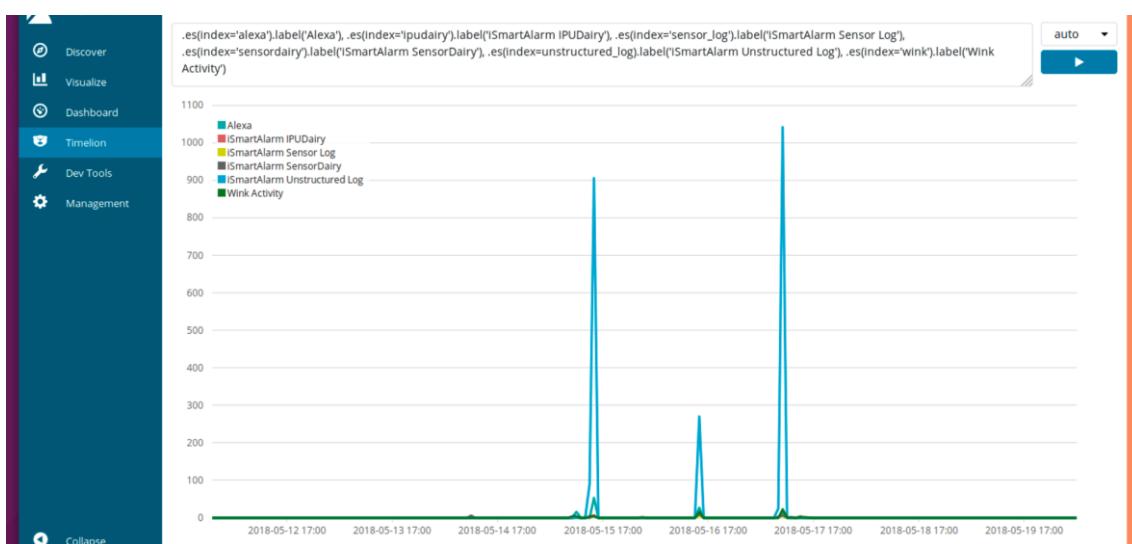


**Figure 55 – Creating index pattern**

Then, you must set Time field at Step 2 to make timeline.

4. Make timeline with command to Timelion like below:

```
.es(index='alexa').label('Alexa'), .es(index='ipudairy').label('iSmartAlarm IPUDairy'),
.es(index='sensor_log').label('iSmartAlarm Sensor Log'),
.es(index='sensordairy').label('iSmartAlarm SensorDairy'),
.es(index='unstructured_log').label('iSmartAlarm Unstructured Log'),
.es(index='wink').label('Wink Activity')
```

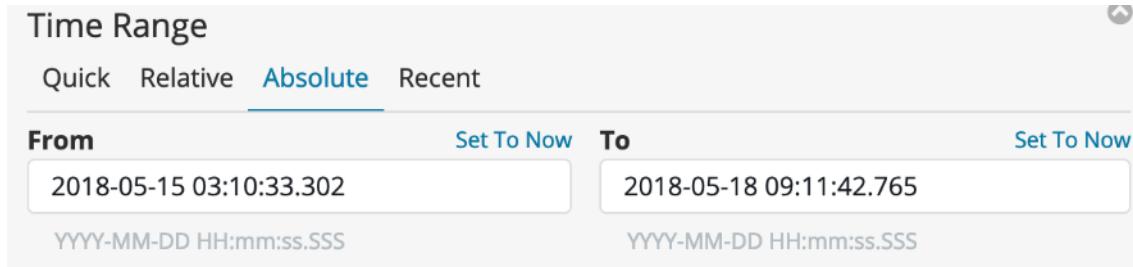


**Figure 56 – Making timeline with Timelion**

5. Save search results to create information for Dashboard.

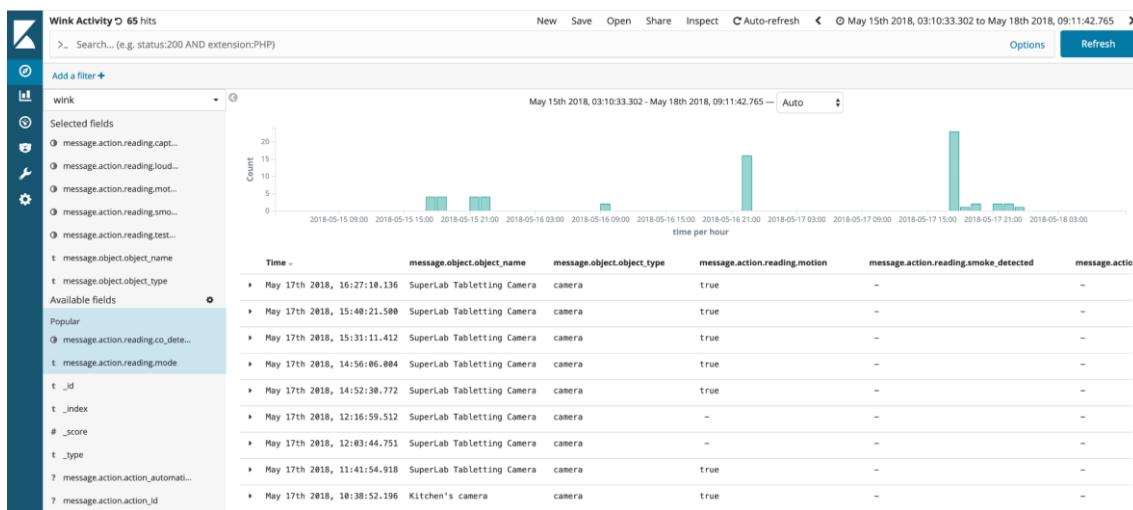
After adding all the indexes, you must set the time range to near the time of the incident before you can

query the information to output the data.



**Figure 57 – Adjusting time range**

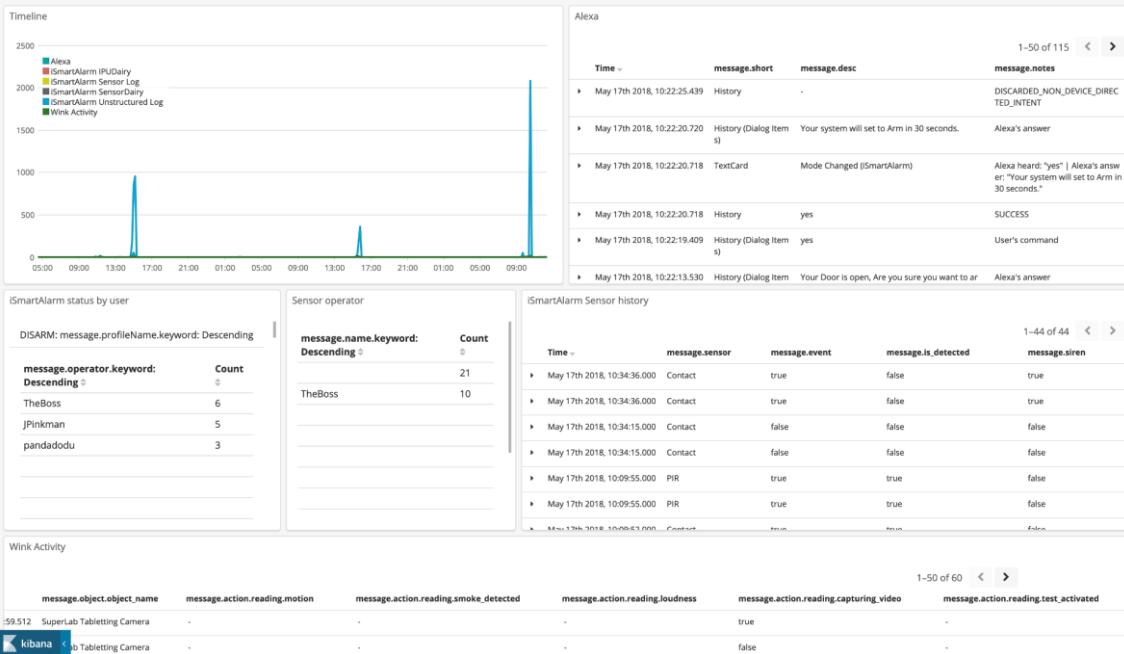
Then make output by choosing necessary fields from each index. If you get desired result, click 'Save' to preserve it.



**Figure 58 – Making organized result by choosing necessary fields from each index**

## 6. Use Dashboard.

In Dashboard, you can integrate various information such as timeline, search result, etc. into one page, by loading each results with 'Add' button.



**Figure 59 – Dashboard window with various information integrated**

## 5 Appendix

### 5.1 List of Collected Evidence Files

Index	File path or Filename	Hash Value (SHA256)
3.1.1	proCDATA.zip	A6419881C0B069633308406CDAD42B34E 360A9F7D5B5878BEAF728C773462E97
	blk0_sda.bin	29E3ECFACCDA09D8FF2026CF71D282C7 026F7D0962C1508CD6F15BEB51EC4AC4
	blk16_sdb.bin	B167346B8EABDA6DD8C7148E4AA15CD5 F62148393ACCB7570E60F2505F010B7B
	blk32_sdc.bin	1301CA17CF7DE7DA7B52AE178C1DBBFE D8FA4BCC97F081C6FB29C981EF43FD0B
3.1.1.1	/[vol EFS]/wifi/.mac.info	EE607D2A53083B324BB95CBD4549D2171 3C725520D6C1FD8F7FDDDB50BB582B44
	/[vol EFS]/bluetooth/bt_addr	2D09571ACF2D00BC1FC9CFA3D3CC0F28 C65C8190D00D7085020B97D76F39C4F9E
	/misc/bluedroid/bt_config.conf	15716FA115113B066250AE30351D425CAA 1F21CF6140A9D20BCD09D1D4FBCAC5
	/property/persist.sys.timezone	650575E62B1F4A4702F14C5160607144F1 6605D06B8C431296CD51CCD54CA79B
	/misc/wifi/wpa_supplicant.conf	3F01016D8B72B9DE65929607B49D308B64 0DCB5CE3F5723EE93426E8FA56E7D5
3.1.1.2	in '/media/0/DCIM/Camera'	
	20180410_092015.jpg	74405FAE9AC96A1919AFF3DD01FA27E14 2DF20FB86AEC10FD1D1B76821B1F03B
	20180410_092059.jpg	34E38BBDEB73B710BAF6493E6CF204905 8027CB56635CE274CCEBBDA49CE63F1
	20180326_164922.jpg	49179327C772E26C4E0BE5F034D9E00575 E887A3E4EAD6D76DD4BB17F9176633
	20180410_091838.jpg	33A4F66603D5334A8881B144B65D3DA448 FE7523B1155CACDFBBBD95D16C9A88
	20180410_091924.jpg	689FAAAA8738473A13BF46762C329F05A6 88B0AB8048D54F93579DFAB8F91CC9
	20180410_092120.jpg	AE1A305E8EDB7B6A25CBF2A3400A81D8 B187A8B1D2D1FADF4C9F5260DB584EA7
	20180410_114114.jpg	E568A3A1008FA606025FDB10A3FEF482C 1CE624845633A1D965F97EB88B5B4C7
	20180410_114127.jpg	8A9A60A403CAD8986E926F4F9728C6A6B 348D07F68FAD01CFD0E2045DE6B2DCA
	20180410_114136.jpg	04C58F062784951758DE6918601EFEE6F5 E4ED9EB71AB8EC30FF3B62B94286CD
	20180410_152808.jpg	E2595361B1D8AFDAD8F85228F2935CD6B BD935DDC1DADB1EB4596C6252A10F88
	20180410_114908.jpg	5E85FA50DEFF028DAFB8F033AB1C206C BC42BE41DFF9A0AAD74B06F48A02A8DA
	20180410_114913.jpg	EE2C3C8E4D7D623D3D3657F1CBD7B437 E095336C4C8EFBD2E3A66DB2821F2081
	20180425_164054.jpg	17261E8B689BBAF6CC8FCC1A7156365E5 C9FCF5CD45BEE8B0249FE72CB22186C
	20180425_164335.jpg	6B17FFB832F016E505BDE0E428A3104D1 19A336FD81D33C17D340683A142A41E
	20180425_164540.jpg	198C3600BE7184185486A82B18DE7A2827 E99C122E26068AE2DF484644795D2E
	20180425_164621.jpg	5E22E6493C846093AB1C1003A9F674078F 74B58DA39E64D57F82FBF8F453E33E
	20180425_165004.jpg	5BBAB45A4EF10EEB43EDE2F8F94562A71 38EF801287343ADF0700CF1658CFC4E

Index	File path or Filename	Hash Value (SHA256)
3.1.1.3	20180425_165025.jpg	6B8C9C3A6AE153339033871EA4D4CCBB B701FC41D03164AB7B19A595809F353E
	20180425_165051.jpg	B6F4B10703C5BD0AEC94FC4F1F4B24D3 6B62964B54136F7DFA1C2B37102C195A
	20180425_173708.jpg	D379283EE696DC905449802B4D951C84F 83ED4B040E31CED36FAB121E6628492
	20180425_181145.jpg	7E86CA6B68B026B713BF79DBC0C8551C7 73D5EB78447AD7112849B3E7853DF80
	20180425_181201.jpg	986A0B9BF426CCA7968CBE3BF5909B618 15AC43F9DBEF762B355266FECB623CF
	20180425_181209.jpg	13534967AE0DB7CB5A9025C890D11BEFF 286D5F8E70D74AF44DCEB067A0A168F
	in '/media/0/DCIM/Screenshots'	
	Screenshot_20180330-203354.png	277682D4A35AF987A06F379FE9A915C09 DD4832E28C9E3795DBAFB4BAC84BCB9
	Screenshot_20180502-132904.png	A1A7007A2BD49555597096A2BE7D3A2AC 82EB2FE68DC137E15A664DB31F428F
	in '/data/iSA.common/'	
3.1.1.3	databases/iSmartAlarm.DB	DBCA18D95BD9E34C122FE957249D2593 C22DFC1AB771C9CF95FCC5DFB9696F2E
	shared_prefs/iSmartAlermData.xml	623D0B870463EB8E770C0821C33C80C8D 567D14B8AD7E050FA7D5CF1E6954BE6
	in '/data/com.quirky.android.wink.wink'	
	databases/persistenceDB	D25074D7A04045B590D9F52045E96A5314 A217DF8AA7839CCC35A11059829EDF
	shared_prefs/wink_local_pref_470654.xml	E3A91E1FF45A2C30AC3251AB2DD603961 CC6EBBE763A699C1F0C6B283330ADC7
	shared_prefs/com.quirky.android.wink.wink_p_references.xml	C56A3739DA0C1842529E63D0A3A3F2DB2 CD4ED06083ADFE569764B0EF4FC403B
	shared_prefs/user.xml	37EEBAB4EEA4F7F0CAD9045472D311AC 81125994B3CBC38E4D34810DC7613F3D
	in '/data/com.quirky.android.wink.wink/cache/image_manager_disk_cache/'	
	3bf47cf7de82576cc12107fd27867f9a847991 ab6bf36c98aaa124e1db888e3a.0	66E68D3A84C8B6D0DCDFB1152F9726616 DF1924B241D2DD147B9E4862F6AC7E0
	ce801688848e08931fd5f5ca8597671920929 09897d2eeeea59390d0773092b84.0	D2A281ADC5019942B9DADFABB813FC61 FCC377B85B13571D3B558D0604487AA2
3.1.1.3	e6decba9336ebb90a64c84579b5be16ebdf6 303408f1a2432d5cf4dba16e1e3e.0	7B7D82357891E227FB671145DA6730B583 2F2F19446A500CA8E3D8791292D6C6
	4bf6c92039987221a8e7a28614e5a216241d 143523535c05651be1977bd90db6.0	41AF1F24A70E878E216AE58DB7FCFCB63 151F6ED1BEAF628247ED8C0BD94C1BF
	in '/data/com.nest.android/ '	
	databases/cache	2B81DC2BCCB44ED398217E1E61CC669A F8C156619962E2E4E176F0EDAB507BA0
	shared_prefs/com.nest.android.preferences.xml	4F941A8BC6EC1AC2C568405C412A005B3 C8238C07C632C52B7502537D88BEE6E
	in '/data/com.nest.android/cache/ '	
	f315c6e2b5434a5381f1f5be 6f73b4b3/frame_database	D2E194174A8BB27271FD33AD517CA4E0A DBE658DB77E0C4785D7C4AB61F3F87B
	video 78 extracted from 'frame_database' (78.mp4)	8E46A4423010C3D2584040D8A5B0C36018 E939F44EC00ADD907ED803A939055C
	video 79 extracted from 'frame_database' (79.mp4)	3B663337C7FD55684EDAB8EEC5B72BFE CB538008CC63B8BFFAC164F23F55A17E
	video 201 extracted from 'frame_database' (201.mp4)	418BA8A6785DF3E2725E71526C2570C7D 1A67615A10E21F9430DB54C3921DC8E

Index	File path or Filename	Hash Value (SHA256)
	video 202 extracted from 'frame_database' (202.mp4)	52A9CCC30A75F0A018EBC9AF3E29DABC 15CE0D161A28179F61DE1A6FAD283C47
	video 203 extracted from 'frame_database' (203.mp4)	97528651708DC654F0E8AEAEADBD15AFE E2830FF54AF94962BB6B0DF1B68011E
	cache/cache-1332523362.json	7DEB3A6A6AEF1813C59405CB1BE264120 0A5B7801F856CE15574089C2B417662
	cache/cache-1503821048.json	EE3CC2F64087BA423E11B3D79632A502B AF8869A0F55D0E5EDA8C79E91BB00FB
	dcnetwork/-532597445-1277665155	C4105AAA74C38C79E1089480E6C619F8B 6631FA519AB7DA02ACFD51921CC73D1
	dcnetwork/-532597445-1563280601	25BEFF1A989D35CD7BED102707E27F553 59ED85E9C2F69474C5468829E34D0AB
	dcnetwork/-532597445-949917945	CC1FF1A5901ADD1BD1A4937DD5082E81 E0AE3D20CDF57B372A95178B13DA7B7F
	in '/data/com.netgear.android/'	
	files/default.realm	62DB5BF9CA65AC0193869D703D2863030 67F9CD743C93964469518FCAADC4423
	shared_prefs/Phoenix.xml	F254FE58C6A4DE3076B77A260F0F073320 CE59F591EAEBB97ACF0803CE40EAE8
	in '/data/com.netgear.android/cache/http/'	
	67f43186731b8ca1b0dbf25ec25c5391.0	043D4512BB9581F3F874AB297421015929 41797E7D44B89F93B5BBA626736287
	995a98df55fff0452ef26af1da49e80d.0	FD0ECD1C016AFC72B0D52D9139C6C293 FFE98FFB0CC89B025771E93CEADC9531
	a52213fe457b9b76431d909d1bce6d7d.0	C218B4847704E36333F3B8AC8DBA90E25 6C29D5D9170E98B4FC5E5B376C2FC5F
	41593f144685f25ba6494ce186d1ff7a.0	CCBDD5CEF12FE6FE26E56AD0ADEE912 7A0060D0D8BA44D1BE6389FC89FFF5926
	a03a80b969307da39f6fc01a2d29af9.0	B4CADE206898F5066F42488FE4C813DFB 19EB02F606DF8E5D6BF76C4356BF34C
	92af7d396df2aa692712d3cedc7ce004.0	79771637F1D9687F76860692F15894A73F CB83413A2D3BAB1DFB56B112EE20D8
	in '/data/com.vestiacom.qbeecamera/shared_prefs/'	
	com.vestiacom.qbeecamera_preferences.xml	6157C38BA11FFFA173C39B7F893E0CF0F 6FC6BD3A415F4FB4671EF9F8B73B97
	in '/data/com.amazon.dee.app/'	
	databases/map_data_storage_v2.db	848AA1653D190E14E1B7AB750FDF082BF 05BC9CED98F203CCE89B736A81F1A55
	shared_prefs/service.identity.xml	6B41587850D2CA2184B54C6217F5529718 FE338E13E11B741986E6F12A3B8993
3.1.3.2	the front 128MB of the memory image	A2BDA9A1F35129DF90D1512F0B9C308F9 7B5FFCF68B6EB8F695985D9E83E5940
3.1.3.2	the back 128MB of the memory image	8DE21D92FFDF97F726BC317098E614AD7 92DAEB3E1A3C2FB7DA72FF000A501B4
3.1.3.3	in '/tmp/media/nand'	
3.1.3.3	eventlog	403CCA8AB68C9CB8F56F80C190D173A20 2758BBD86B738DFD15F07006F0B336D
3.1.3.3	vzdaemon/conf/automation/automation.map	9F02497C471A0AC63EF94178765F5AA3A F264FF3FF02ABBEF64A578A6EE73C9
3.1.3.3	vzdaemon/conf/cameras/59U17B7BB8B46.json	A968314F2F1FD4919B240790898201F740 E6C6A5CC6E4A4A8652A0E82254CDE4
3.1.3.3	vzdaemon/conf/BasestationConfiguration.json	11C42A1437C8960D1F12B4E53C43D2DB4 C703F728EC3EDC0E60F16274CC740D2
3.1.4	/root/wifi/nvram.conf	1A3D1D0E59211F1FF1C1D984560FAF0AA FB6D9A77F52F2D1AA87B80C031AAECD
3.1.4	/database/bd_addr	44D21D01D92906F1CE32F327EE830D5EA 95BD443374CFD9E9AF7DAAB471CDD7

Index	File path or Filename	Hash Value (SHA256)
3.1.5	/tmp/all.log	A0E6232706712D3EB5A152A3D16C8D668 65934C22F4FA210B2C2D501097BA6C0
	/database/zigbee-version.txt	2FE650FFD90B43CA35D6192FA42700C6D 85697D00F60FC09982BD0DAC68AFC5C
	/database/authorized_keys	93858A6990B440044B26E1EF327D72246D E1DC5C0BE573640D6DAA2379D5CB70
	/database/wpa_supplicant.conf	E6BE6B2AD3DE42A1339E494C50A4F6A6 A7E697254783349EC08D2DCB050D33C0
3.1.5	cift_amazon_alexa.db	CC25ACAAC221DB6FB17AAAA59409DA94 A7D178DF126B72E7EFAA7A301D85D0A6
	cift_amazon_alexa_ACCOUNT.csv	19841700F182A6DC5F3CCA1553723CD08 F2B3FF571997B5538B533B04C9FA8D2
	cift_amazon_alexa_ACQUIRED_FILE.csv	6BA855F84ED9D4AE0B063FB5B73F6E607 65C2CDA17F3435811049E112F65A074
	cift_amazon_alexa_ALEXA_DEVICE.csv	FB38ED69E159989F027BAA28424230010A A174D3DA53AFF46E5230DB0177E196
	cift_amazon_alexa_SETTING_MISC.csv	EF88CB147057A8BEBFA21954E0E71D800 46E806ADA6BAA824A1CD00ACCED5BDB
	cift_amazon_alexa_SETTING_WIFI.csv	4D81E394EF95DB0194F881011684682002 98C0F404B0950485DAC71DCA767540
	cift_amazon_alexa_SKILL.csv	90E70A6F79297528D51AE0FF9232BCE2C CA9C0E9890603A453088AE10907974A
	cift_amazon_alexa_TIMELINE.csv	FB2254AB38D7FD4CE2B5F0FF55AF1D77 AF28BEB6813142E3F4A65EE855CCA4BE
	last_progress_log.txt	C8D4FC273AAB25DD9F32DBC9C365C378 116B2451AAD896D0A79D6981BC89DA47

## 5.2 Additional Explanation of the Forensic Analysis Details

### 5.2.1 Jessie Pinkman's Smartphone

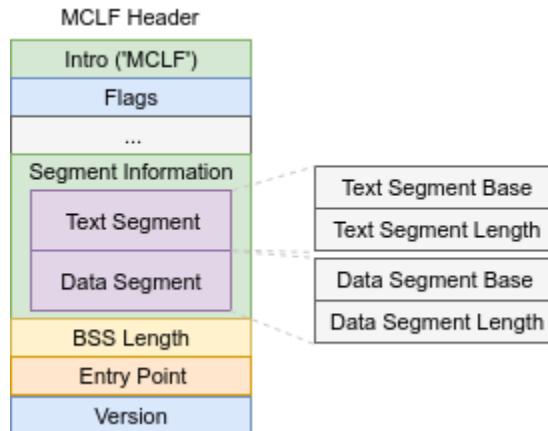
#### 5.2.1.1 Bootloader Data

- 'blk16\_sdb.bin' file: sboot.bin
  - The latest smartphones, including the Samsung Galaxy 6 series, include a bootloader called 'sboot'.
  - sboot includes a part of MobiCore Load Format (signature MCLF) to be run under the TrustZone, and we found the signature in the file as below.

```
root@kali: ~/Desktop# hexdump -C blk16_sdb.bin | grep MCLF
000af590 4d 43 4c 46 03 00 02 00 01 00 00 00 02 00 00 00 | MCLF..... |
0013fc40 05 00 02 00 08 10 d0 07 09 00 b0 40 4d 43 4c 46 | .....@MCLF|
00159000 4d 43 4c 46 05 00 02 00 0b 00 00 00 01 00 00 00 | MCLF..... |
0016b000 4d 43 4c 46 05 00 02 00 08 00 00 00 02 00 00 00 | MCLF..... |
```

Figure 60 – Search result in 'blk16\_sdb.bin' for the signature 'MCLF'

- The format of MCLF file is shown in the below figure.



**Figure 61 – The format of MCLF file**

- If you look at some of the MCLF data, you can see the strings related to booting as follows.

000A35C0	53 4D 2D 47 39 32 35 46 00 00 00 00 00 00 00 00	SM-G925F.....
000A35D0	45 58 59 4E 4F 53 37 34 32 30 00 00 00 00 00 00	EXYNOS7420.....
000A35E0	25 73 3A 20 50 57 52 20 4B 65 79 20 69 73 20 6E	%s: PWR Key is n
000A35F0	6F 74 20 72 65 6C 65 61 73 65 64 2E 0A 00 00 00	ot released.....
000A3600	46 61 69 6C 20 74 6F 20 6C 6F 61 64 20 53 65 63	Fail to load Sec
000A3610	75 72 65 20 50 61 79 6C 6F 61 64 21 21 20 5B 72	ure Payload!! [r
000A3620	65 74 20 3D 20 25 78 5D 0A 00 00 00 00 00 00 00	et = %x].....
000A3630	6C 6F 61 64 20 53 65 63 75 72 65 20 50 61 79 6C	load Secure Payl
000A3640	6F 61 64 20 64 6F 6E 65 2E 0A 00 00 00 00 00 00	oad done.....
000A3650	0A 43 68 69 70 20 49 44 20 3A 20 25 30 34 78 25	.Chip ID : %04x%

**Figure 62 – Boot-related strings found in MCLF data area**

- ‘blk32\_sdc.bin’ file: cm.bin
  - This file is implemented by sboot bootloader, and seems to be related to anti-flashing purpose.
  - It contains data related to ‘CryptoManager’, name of which implies that it is for hardware security platform providing cryptographic function and key management.

00000000	1E AB AD 10 00 00 20 00 E8 00 00 04 70 00 00 04	.«.... .è....p....
00000010	40 00 00 04 68 00 00 04 00 10 00 82 27 10 15 20	@....h.....,'..
00000020	E8 01 00 04 00 00 00 00 00 00 00 00 00 00 00 00	è.....
00000030	43 72 79 70 74 6F 4D 61 6E 61 67 65 72 56 32 30	CryptoManagerV20
00000040	40 00 00 F0 00 4C 44 B9 1F 04 00 71 A0 00 00 54	@..8.LD^...q ...
00000050	40 00 00 F0 00 5C 44 B9 1F 04 00 71 E0 17 9F 1A	@..8.\D^...qà.Ý.
00000060	00 7C 40 93 C0 03 5F D6 00 00 80 D2 C0 03 5F D6	. @“À._Ó..€ÒÀ._Ó

**Figure 63 – String ‘CryptoManagerV20’ found in ‘blk32\_sdc.bin’**

### 5.2.1.2 ‘frame\_database’ of ‘Nest’ Application

This section describes the structure of ‘frame\_raw\_data\_table’ table of ‘frame\_database’ file in ‘Nest’ application data.

A video, which is concatenation of successive still pictures, can be generated by combining frames with same chunk ID. SPS (Sequence Parameter Set) and PPS (Picture Parameter Set) information are stored in the record of the first frame.

Each frame record has timestamp so we can know when an individual still picture was taken.

The following table describes the roles of the fields of ‘frame\_raw\_data\_table’ table.

**Table 40 – Fields of ‘frame\_raw\_data\_table’ table in ‘Nest’ application**

Field Name	Description
frame_time	Timestamp of frame
chunk_id	Frames in the same video have the same chunk ID.
chunk_version	(Null)
gop_start_rowid	Timestamp of the first frame, which has ‘sps_bytes’ and ‘pps_bytes’, of a video
sps_bytes	SPS (Sequence Parameter Set) data
pps_bytes	PPS (Picture Parameter Set) data
frame_bytes	Frame data encoded with H.264 codec
chunk_complete	If the value is set to 1, it means that the record is the last frame of a video.

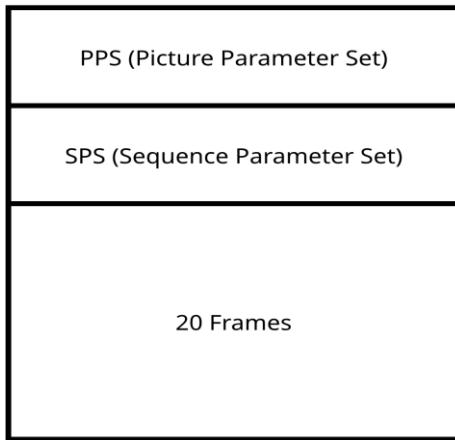
The figure below shows some records of ‘frame\_raw\_data\_table’ table.

	frame_time	chunk_id	chunk_version	gop_start_rowid	sps_bytes	pps_bytes	frame_bytes	chunk_complete
1	1526289810036	1526288836	NULL	-1	BLOB	BLOB	BLOB	0
2	1526289874202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
3	1526289942202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
4	1526290010202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
5	1526290034202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
6	1526290106202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
7	1526290178202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
8	1526290254202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
9	1526290326202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
10	1526290398202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
11	1526290474202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
12	1526290546202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
13	1526290622202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
14	1526290634202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
15	1526290706202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
16	1526290778202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
17	1526290854202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
18	1526290926202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
19	1526290998202	1526288836	NULL	1526289810036	NULL	NULL	BLOB	0
20	1526291104536	1526288836	NULL	1526289810036	NULL	NULL	BLOB	1

**Figure 64 – Records of ‘frame\_raw\_data\_table’ table in ‘Nest’ application**

We can extract and play a video file from this database by combining frame bytes of successive 20 frames with the SPS and PPS bytes in the first frame as shown in the below figure and converting into mp4 file with ffmpeg library. Also we can separate the video into individual frames with timestamp marked.

### Nest Cached Video H.264



**Figure 65 – Video file structure extracted from ‘frame\_raw\_data\_table’ table in ‘Nest’ application**

## 5.2.2 iSmartAlarm Base Station

In this part, we explain more details of the format of record in the diagnostic logs.

Unstructured Log contains various values including the signature of the file, and Sensor Log contains a record of the contact sensor and the PIR sensor (motion sensor).

### 1) Unstructured Log

Unstructured Log can be divided into Header, Identifier Table, Log Chunks, and their approximate formats are as follows.

- Header
  - Header is the area at the beginning of the diagnostic log, which contains various values including the signature of the file.
  - We cannot tell you exactly how the unknown fields are used for, unless directly analyzing the logging executables in the device.
  - The following table shows the currently identified format of Header.

**Table 41 – The format of Unstructured Log Header**

Offset	Field Name	Size	Specified Value	Description
0x00	Stream Signature	8 Bytes	“LOGTG\x00\x00\x00”	Signature of unstructured log
0x08	Unknown 1	4 Bytes	-	-
0x0C	Unknown 2	4 Bytes	-	-
0x10	Unknown 3	4 Bytes	-	-
0x14	Unknown 4	8 Bytes	-	-
0x1C	Unknown 5	4 Bytes	“star”	-

- Identifier Table
  - Identifier Table contains all the identifier values contained in Log Chunks.

- The table signature included in the format is also similar to the identifier, which is also included in Identifier Chunk.
- It is assumed that ‘Unknown’ field of Identifier Chunk has a relative value such as offset in order to refer to the Log Chunk having identifier value.
- However, since it can have same value with other Unknown field, we can guess that its value is as the type or property of Log Chunk.
- The following table shows the currently identified format of Identifier Table.

**Table 42 – The format of Unstructured Log Identifier Table**

Offset	Field Name		Size	Specified Value	Description
0x00	Table Signature		4 Bytes	-	Signature of Identifier Table
0x04	Unknown Area		24 Bytes	-	
0x1C	Identifier Chunk N	Identifier	4 Bytes	-	Identifier used for Log Chunk [assumption]
		Unknown	2 Bytes	-	
-	Identifier Chunk N+1	Identifier	4 Bytes	-	
		Unknown	2 Bytes	-	

- Log Chunk
  - Log Chunk starts from offset 0x60000, and several chunks are connected in succession.
  - The contents of the data are presumed to be associated with ‘Data Type’ or ‘Identifier’ field.
  - The following table shows the currently identified format of Log Chunk.

**Table 43 – The format of Unstructured Log Chunk**

Offset	Field Name		Size	Specified Value	Description
0x00	Chunk N	Signature	2 Bytes	“\$@”	Signature of Log Chunk
0x02		Data Size	4 Bytes	-	Size of ‘Data’
0x06		Separator 1	2 Bytes	“..”	
0x08		Identifier	4 Bytes	-	Identifier of Data [assumption]
0x0C		Separator 2	2 Bytes	“..”	
0x0E		Data Type 1	2 Bytes	-	[assumption]
0x10		Data Type 2	N Bytes	-	[assumption]
-		Separator 3	2 Bytes	“..”	
-		Data	N Bytes	-	N = value of ‘Data Size’
...		Chunk N+1	...	...	...

## 2) Sensor Log

Sensor Log contains a record of contact sensor and motion sensor. event classification for each sensor is available in ‘Data Type’ field. When an event occurs, it is stored in the log as ‘ALARMDOOR’ (contact sensor) and ‘ALARMPIR’ (motion sensor), respectively. Value in the ‘Data’ field can be general text or JSON format data. For general text, it is simply used to record the current event, and JSON data is

used to send POST request to the cloud service of iSmartAlarm. This JSON data allows you to get a short description of the current event.

Examples of sensor log record are shown in the below table.

**Table 44 – Examples of Sensor Log records**

\$_@000000005AFD3E9C::ALARMDOOR::door is open, and send to cloud
\$_@000000005AFD3E9C::ALARMDOOR::{"SensorID":"000A8540","MessageType":"0","TS":"1526546076051","IsAlarm":"1","fileGroup":"0","ProfileId":"2","ModelId":"2","DetectAlarm":"0"}
\$_@000000005AFD38D3::ALARMPIR::pir is triggered, and send to cloud
\$_@000000005AFD38D3::ALARMPIR::{"SensorID":"0006B4E5","MessageType":"0","TS":"1526544595279","IsAlarm":"1","fileGroup":"0","ProfileId":"0","ModelId":"0","DetectAlarm":"1"}

The following table shows the currently identified record format of Sensor Log.

**Table 45 – The format of Sensor Log record**

Offset	Field Name	Size	Specified Value	Description
0x00	Log Signature	2 Bytes	"\$_@"	Signature of record
0x02	Timestamp	16 Bytes	-	Hexadecimal string of epoch timestamp Ex) 000000005AFAAABB In decimal, 1526377147 In epoch time, May 15 2018 AM 9:39:07 UTC
0x12	Separator 1	2 Bytes	".."	
0x14	Data Type	N Bytes	-	[assumption]
-	Separator 2	2 Bytes	".."	
-	Data	N Bytes	-	Description follows at below

Here is the description of the ‘Data’ field in records with ‘ALARMDOOR’ and ‘ALARMPIR’ data type.

- In Case of General Text Format

Data Type	Event	Value of ‘Data’
ALARMDOOR (Contact Sensor)	Door Open	door is open, and send to cloud
	Door Closed	door is closed, and send to cloud
ALARMPIR (Motion Sensor)	Motion Detected	pir is triggered, and send to cloud

- In Case of JSON Format

In case of JSON format, there are 8 key-value pairs contained in JSON object.

Key	Description	Example	
		Data Type	Value
SensorID	ID of Sensor	ALARMDOOR	'000A8540'
		ALARMPIR	'0006B4E5'
MessageType	Type of Event	ALARMDOOR	Door Open: '0' Door Closed: '1'
		ALARMPIR	Motion Detected: '0'

TS	The Time Event Happened	*	Epoch Timestamp. ex) '1526544595279' (2018-05-17 08:09:55 UTC)
IsAlarm	-	*	Always '1'
fileGroup	-	*	Always '0'
ProfileId	iSmartAlarm System Mode Profile	*	If iSmartAlarm is in: ARM mode: '0' / HOME mode: '1' / DISARM mode: '2' / (PANIC mode: '3')
ModelId	Same As 'ProfileId'	*	Same As 'ProfileId'
DetectAlarm	Whether The Alarm Has Set	ALARMDOOR	Door Opened with 'HOME' mode or 'ARM' mode: '1' / Else: '0'
		ALARMPIR	Motion Detected with 'ARM' mode: '1' / Else: '0'

## 6 References

- [1] "Autopsy," [Online]. Available: <https://www.sleuthkit.org/autopsy/>.
- [2] "ReFirmLabs/binwalk: Firmware Analysis Tool," [Online]. Available: <https://github.com/ReFirmLabs/binwalk>.
- [3] "DB Browser for SQLite," [Online]. Available: <https://sqlitebrowser.org/>.
- [4] "Wireshark · Go Deep.," [Online]. Available: <https://www.wireshark.org/>.
- [5] "010 Editor - Professional Text/Hex Editor with Binary Templates," [Online]. Available: <https://www.sweetscape.com/010editor/>.
- [6] "Realm Studio: open, edit, and manage your Realm data," [Online]. Available: <https://realm.io/products/realm-studio/>.
- [7] "Nest Home Simulator | Nest Developers," [Online]. Available: <https://developers.nest.com/guides/home-simulator>.
- [8] "Online JSON Viewer," [Online]. Available: <http://jsonviewer.stack.hu/>.
- [9] "FFmpeg," [Online]. Available: <https://www.ffmpeg.org/>.
- [10] "Welcome to Python.org," [Online]. Available: <https://www.python.org/>.
- [11] "Enterprise Application Container Platform | Docker," [Online]. Available: <https://www.docker.com/>.
- [12] "Open Source Search & Analytics · Elasticsearch | Elastic," [Online]. Available: <https://www.elastic.co/>.
- [13] "Kibana: Explore, Visualize, Discover Data | Elastic," [Online]. Available: <https://www.elastic.co/products/kibana>.
- [14] "dex2jar download | SourceForge.net," [Online]. Available: <https://sourceforge.net/projects/dex2jar/>.
- [15] "Samsung Galaxy S6 edge SM-G925F specs," [Online]. Available: <https://www.sammobile.com/devices/galaxy-s6-edge/specs/SM-G925F/>.
- [16] "Information for Europe/Zurich - TimeZoneDB," [Online]. Available: <https://timezonedb.com/time-zones/Europe/Zurich>.
- [17] "How do the Arm, Home, Disarm modes work? – Support Center," [Online]. Available: <https://support.ismartalarm.com/hc/en-us/articles/204821968-How-do-the-Arm-Home-Disarm-modes-work->.
- [18] "[CVE-2018-16222 to 16225] Multiple Vulnerabilities in QBee and iSmartAlarm Products – Francesco Servida's Blog," [Online]. Available: <https://blog.francescoservida.ch/2018/10/31/cve-2018-16222-to-16225-multiple-vulnerabilities-in-qbee-and-ismartalarm-products/>.
- [19] "Realm Database," [Online]. Available: <https://realm.io/products/realm-database/>.

- [20] [Online]. Available: <https://alexa.amazon.com/spa/index.html>.
- [21] "iSmartAlarm - iSmartAlarm Home Security System Owner's Manual," [Online]. Available: [http://www.produktinfo.conrad.com/datenblaetter/1400000-1499999/001424846-an-01-en-ISMARTALARM\\_TUER\\_\\_UND\\_FENSTERKONTAKT.pdf](http://www.produktinfo.conrad.com/datenblaetter/1400000-1499999/001424846-an-01-en-ISMARTALARM_TUER__UND_FENSTERKONTAKT.pdf).
- [22] "What are the features and specifications of the Arlo Wire-Free system?," [Online]. Available: <https://kb.arlo.com/1968/What-are-the-features-and-specifications-of-the-Arlo-Wire-Free-system>.
- [23] "Wink | Wink Hub," [Online]. Available: <https://www.wink.com/products/wink-hub/#specs>.
- [24] "Wink Hub - WikiDevi," [Online]. Available: [https://wikidevi.com/wiki/Wink\\_Hub](https://wikidevi.com/wiki/Wink_Hub).
- [25] "[CVE-2018-16225] Public Disclosure – QBee Camera Vulnerability – Francesco Servida's Blog," [Online]. Available: <https://blog.francescoservida.ch/2018/09/16/cve-2018-16225-public-disclosure-qbee-camera-vulnerability/>.
- [26] "QBee Network Communication Vulnerability - YouTube," [Online]. Available: [https://www.youtube.com/watch?v=dd8vt0\\_DJF4](https://www.youtube.com/watch?v=dd8vt0_DJF4).
- [27] "iSmartAlarm - Wikipedia," [Online]. Available: <https://en.wikipedia.org/wiki/ISmartAlarm>.