Computers & Security 94 (2020) 101833

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

What's really '*Happn*ing'? A forensic analysis of Android and iOS *Happn* dating apps

Shawn Knox, Steven Moghadam, Kenny Patrick, Anh Phan, Kim-Kwang Raymond Choo*

Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249, USA

ARTICLE INFO

Article history:
Received 3 January 2020
Revised 1 April 2020
Accepted 7 April 2020
Available online 28 April 2020

Keywords:
Dating apps
Online dating
Dating app artifacts
Mobile forensics
iOS forensics
Android forensics

ABSTRACT

With today's world revolving around online interaction, dating applications (apps) are a prime example of how people are able to discover and converse with others that may share similar interests or lifestyles, including during the recent COVID-19 lockdowns. To connect the users, geolocation is often utilized. However, with each new app comes the possibility of criminal exploitation. For example, while apps with geolocation feature are intended for users to provide personal information that drive their search to meet someone, that same information can be used by hackers or forensic analysts to gain access to personal data, *albeit* for different purposes. This paper examines the *Happn* dating app (versions 9.6.2, 9.7, and 9.8 for iOS devices, and versions 3.0.22 and 24.18.0 for Android devices), which geographically works differently compared to most notable dating apps by providing users with profiles of other users that might have passed by them or in the general radius of their location. Encompassing both iOS and Android devices along with eight varying user profiles with diverse backgrounds, this study aims to explore the potential for a malicious actor to uncover the personal information of another user by identifying artifacts that may pertain to sensitive user data.

1. Introduction

Dating application (apps) have a variety of functions for users to match and meet others, for example based on their interest, profile, background, location, and/or other factors using functions such as location tracking, social media integration, user profiles, chatting, and so forth. Depending on the type of app, some will focus more heavily on certain functions over another. For example, geolocation-based dating apps allow users to find dates within a certain geographic area (Attrill-Smith and Chris, 2019; Sumter and Vandenbosch, 2019; Yadegarfard, 2019), and a number of dating apps have reportedly "rolled out functionality and pricing changes to help people connect more deeply without meeting in person" in the recent lockdowns due to COVID-19[1]. Popular apps such as *Tinder* allow users to restrict the range to a specified radius, but *Happn* takes this approach a step further by tracking users who have crossed paths. From there, the user can view brief descriptions, pictures or other information uploaded by the user. While this is a convenient way of connecting strangers (Sumter and Vandenbosch, 2019; Veel and Thylstrup, 2018), it could make *Happn* users more vulnerable to predatory behavior, such as stalking (Lee, 2018; Murphy, 2018; Scannell, 2019; Tomaszewska and Schuster, 2019). In addition, it was recently reported that activities on popular dating apps appeared to have increased in the recent COVID-19 lockdowns, as more users are staying and working from home[2]. Such increased usage could have security and safety implications (Lauckner et al., 2019; Schreurs et al., 2020).

Given the popularity of dating apps and the sensitive nature of such apps, it is surprising that forensic studies of dating apps is relatively understudied in the broader mobile forensic literature (Agrawal et al., 2018; Barmpatsalou et al., 2018) (see also Section 2). This is the gap we seek to address in this paper.

In this paper, we highlight the potential for malicious actors to uncover the personal information of other users through a forensic analysis of the app's activity on both Android and iOS devices, using both commercial forensic tools and freely available tools. To ensure repeatability and reproducibility, we describe our research methodology, which includes the creation of profiles, capturing of network traffic, acquisition of device images, and backing up of iOS

* Corresponding author.
E-mail addresses: shawnpknox11@gmail.com (S. Knox), steven@neaxtech.com (S. Moghadam), brian.patrick@utsa.edu (K. Patrick), anh.phan@utsa.edu (A. Phan), raymond.choo@fulbrightmail.org (K.-K.R. Choo).

[1] https://www.pcmag.com/news/how-covid-19-is-changing-dating-apps-and-relationships, last accessed April 29, 2020.

[2] https://www.forbes.com/sites/johnscottlewinski/2020/04/06/covid-19-drives-up-use-of-schedule-based-dating-app-flutter/ and https://www.wired.com/story/dating-apps-coronavirus-covid-19/, last accessed April 29, 2020.

https://doi.org/10.1016/j.cose.2020.101833
0167-4048/© 2020 Elsevier Ltd. All rights reserved.

devices with iTunes (see Section 3). For example, devices are imaged if possible, and iTunes backups are utilized instead for the iOS devices that could not be jailbroken. The images and backups are then analyzed to reveal further artifacts. The findings are then reported in Section 4. This section covers various artifacts recovered from network traffic and files left on the devices from the app. These artifacts are separated into ten different categories, whose data sources include captured network traffic, disk images from the devices, and iTunes backup data. Complications encountered during the study are discussed in Section 5.

Next, we will revisit the extant literature relating to mobile forensics. In these related works, some focus on dating apps (one also covers *Happn*) and others taking a broader approach. The studies discuss artifact collection (from files on the device as well as from network traffic), triangulation of user locations, discovery of social relationships, and other privacy concerns.

2. Related literature

The amount of literature focused on discovering forensic artifacts from both mobile dating apps and apps in general has grown gradually (Cahyani et al., 2019; Gurugubelli et al., 2015; Shetty et al., 2020), although it pales in comparison to other areas of mobile forensics (Anglano et al., 2020; Barmpatsalou et al., 2018; Kim and Lee, 2020; Zhang and Choo, 2020). Atkinson et al. (2018) demonstrated how mobile apps could broadcast personal information through wireless networks despite the encryption standards implemented by apps, such as *Grindr* (a popular dating app). By using a live detection program that takes the network activity of the previous 15 s on a device to predict the app and its activity, they were able to estimate the personal characteristics of various test personas. One was identified as most likely wealthy, gay, male and an anxiety sufferer from the traffic patterns created by opening apps such as *Grindr, M&S*, and *Anxiety Utd* – all discovered despite the use of encryption.

Kim et al., 2018 detected software vulnerabilities in the assets of Android dating apps – user profile and location information, user credentials, and chat messages. By sniffing the network traffic, they were able to find a number of artifacts, such as user credentials. Four apps stored them in their shared preferences while one app stored them as a cookie, all of which were retrievable by the authors. Another was the location and distance information between two users where in some dating apps, the exact distance can be extracted from the packets. If an attacker obtains 3+ distances between his/her coordinates and the victim's, a process known as triangulation could be done to find the victim's location. In another study, Mata et al., 2018 carried out this process on the *Feeld* app by extracting the distance between the adversary and the target, drawing a circle where the distance acted as the radius at the adversary's current coordinates, and then repeating the process at 2+ alternate locations. Once the circles were drawn, the target's accurate location was discovered.

In the realm of general mobile apps, Sudozai et al. (2018) focused on acquiring forensic artifacts for a messaging app on Android and iOS devices. Specifically, by accessing both devices' system files, the following artifacts were obtained – user credentials, non-encrypted images and videos, chats in plaintext, contacts and phone numbers, emails, and audio files. Since most of the data were not encrypted, there are privacy implications, for example should a data leakage occurs due to the use of the app. Jadhav Bhatt et al. (2018) also looked at messaging and social apps on iOS devices by sniffing their network traffic and found that 15 out of the 20 apps sent unencrypted or partially encrypted traffic. Some artifacts found were location data, text messages, emails, and information on searched users on with their Facebook profile links.

Another study focused specifically on the *Happn* app and how one can use it to uncover social relationships was able to deanonymize users through triangulation and piece together details of any profile (Di Luzio et al., 2018). In addition, Di Luzio et al. (2018) explained how an attacker could target a large population by uncovering "their daily routines, their home and work addresses, where they like to hangout on weekends, and who their friends, their relatives, and their colleagues are." They were able to accomplish this on a population of almost 10,000 *Happn* users in Rome, Italy by placing 20 test accounts in popular places in the city and then repeatedly retrieving the list of nearby users while also rearranging the accounts to pinpoint the positions of those users. By using just the data provided by *Happn* to actively track the users, the authors found that an attacker could not only uncover the target's routine or favorite places, but also use that information to discover his/her social network – friends, colleagues, relatives, and so forth.

3. Experiment setup

Happn allows the use of a Facebook account or phone number to sign up and login and requires a recovery email address in case a user cannot login. To retrieve as many forensic artifacts as possible, four Android and four iOS mobile devices with their respective phone numbers were used to create eight profiles that were then split into four pairs of two "romantic partners." The four couples would then conduct common activity over three separate sessions, such as sending messages and rejecting other profiles.

Tables 2 and 3 summarize these eight test devices and the app versions, respectively. Each pair involved a younger and older individual with some of these individuals providing an age on the dating app. This would portray a realistic scenario where many dating apps require a specific age to sign up or an individual wishes to be deceptive. The details of the profiles are shown in Table 4. Now, the experiment workflow/setup will be explained, but is also summarized in Fig. 2.

Table 1
Related literature summary.

		Artifact Types							
		Messages	Images	Location	Credentials	Authentication token	Personal Information	Matches	Matches' information
Studies	Atkinson et al. (2018)						√		
	Kim et al., 2018	√		√	√		√		
	Mata et al., 2018	√	√	√			√	√	√
	Sudozai et al. (2018)	√	√		√				
	Jadhav Bhatt et al. (2018)	√	√	√	√		√		
	Di Luzio et al. (2018)			√			√	√	√

Table 2
Test devices.

Device	OS version	Model
iPhone 6s	iOS 12.4.1	A1688
iPhone 6	iOS 12.4.1	A1549
iPhone 7	iOS 11.2.2	A1778
iPhone 7	iOS 12.1.1	A1660
Galaxy S6 Edge	Android v7.0	Samsung-SM-G925A
Galaxy Tab A6	Android v5.1.1	SM-T280
Galaxy S5	Android v6.0.1	SM-G900P
MEmu Emulator (Galaxy S7 Edge)	Android v5.1.1	SM-G935F

Table 3
Happn app versions.

OS	App version	Installation date
iOS	9.6.2	08/01/19
	9.7	10/30/19
	9.8	11/6/19
Android	3.0.22	08/01/19
	24.18.0	10/30/19

Table 4
Test profiles.

Name	Real age	Online age	Gender	Sexual orientation	Unsolicited messages
Jeffrey Lamcee	58	23	M	Straight	None
Samantha Kasper	20	X	F	Bisexual	Multiple Messages
Joseph Jordan	39	X	M	Straight	None
Dottie Harford	13	22	F	Straight	Multiple messages
Trayvon Jackson	63	X	M	Bisexual	None
Ashley Gray	18	X	F	Bisexual	Multiple messages
Nate Trujillo	18	X	M	Bisexual	None
Haley Thomas	50	X	F	Bisexual	None

X = Online age is the same as Real Age. Unsolicited Messages = messages from profiles outside experiment

3.1. Fiddler proxy

In order to retrieve network traffic between the mobile devices and *Happn*'s server, Fiddler – a web debugging proxy – was utilized to decrypt HTTPS traffic from the iOS devices and was installed on a laptop running Windows 10. Changes were made in the Fiddler settings to allow remote connections and enable the decryption of HTTPS traffic.

Several steps were taken on the client devices to complete the decryption process. After the clients and the Fiddler server were connected to the same wireless network, the clients were configured to use Fiddler as a proxy. Each client downloaded Fiddler's root certificate from http://ipv4.fiddler:8888, a web page served by the Fiddler proxy. After the certificate was installed on each client device, decrypted HTTPS traffic was visible through Fiddler. Once the setup was complete, six sessions involving common activity on the app for each profile/device were conducted, as can be seen in Fig. 3.

While the proxy was able to decrypt traffic for iOS devices, it did not work with Android devices. As an alternative, an Android packet sniffer app, Packet Capture, was run every time there were interactions between the device and *Happn*. Packet Capture has a feature to capture traffic from a single app, making it easy to isolate activity from *Happn*. Fig. 1 shows the complete setup for the capture of network traffic.

3.2. Data acquisition

Besides network traffic, forensic artifacts can be found by acquiring data from the device itself. Logical images were acquired for iOS and Android through the tool, MOBILedit Forensic Express, but it was realized that the images did not contain any new or significant information. For iOS devices, iTunes backups were made after each session so that the application's property lists (plist) and SQL databases could be analyzed. The images from the MEmu emulator were stored in

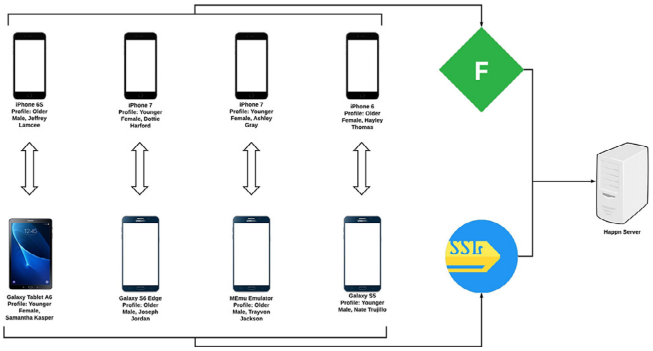

F
iPhone 6S
Profile: Older Male, Jeffrey Lamcee
iPhone 7
Profile: Younger Female, Dottie Harford
iPhone 7
Profile: Younger Female, Ashley Gray
iPhone 6
Profile: Older Female, Hayley Thomas
Galaxy Tablet A6
Profile: Younger Female, Samantha Kasper
Galaxy S6 Edge
Profile: Older Male, Joseph Jordan
MEmu Emulator
Profile: Older Male, Trayvon Jackson
Galaxy S5
Profile: Younger Male, Nate Trujillo
Happn Server

Fig. 1. Device setup

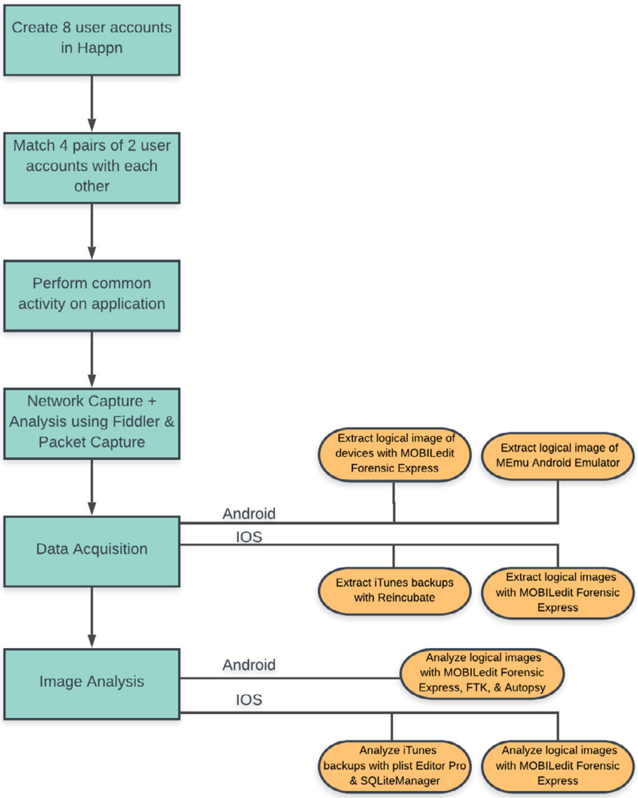

Create 8 user accounts in Happn
Match 4 pairs of 2 user accounts with each other
Perform common activity on application
Network Capture + Analysis using Fiddler & Packet Capture
Data Acquisition
Android
IOS
Extract logical image of devices with MOBILedit Forensic Express
Extract logical image of MEmu Android Emulator
Extract iTunes backups with Reincubate
Extract logical images with MOBILedit Forensic Express
Image Analysis
Android
IOS
Analyze logical images with MOBILedit Forensic Express, FTK, & Autopsy
Analyze iTunes backups with plist Editor Pro & SQLiteManager
Analyze logical images with MOBILedit Forensic Express

Fig. 2. Experiment flow

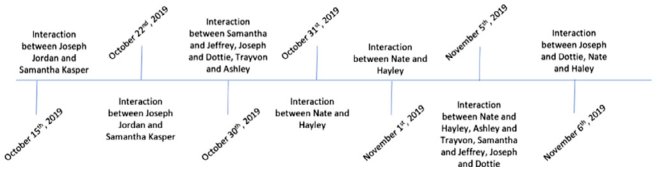

Interaction between Joseph Jordan and Samantha Kasper
October 15th, 2019
October 22nd, 2019
Interaction between Joseph Jordan and Samantha Kasper
Interaction between Samantha and Jeffrey, Joseph and Dottie, Trayvon and Ashley
October 30th, 2019
October 31st, 2019
Interaction between Nate and Hayley
Interaction between Nate and Hayley
November 1st, 2019
November 5th, 2019
Interaction between Nate and Hayley, Ashley and Trayvon, Samantha and Jeffrey, Joseph and Dottie
Interaction between Joseph and Dottie, Nate and Haley
November 6th, 2019

Fig. 3. Sessions timeline

Table 5
Tools used.

Tool	Version	Purpose
MOBILedit Forensic Express	7.0.2.16772	Logical Acquisition + Analysis
FTK	6.0.3.5	Analysis of MEmu VMDK Image
Autopsy	4.12.0	Analysis of MEmu VMDK Image
MEmu Android Emulator	7.0.1	Logical Acquisition + Android Emulator
Reincubate iPhone Backup Extractor	7.7.0.2112	iTunes Backup extraction of iOS devices
plist Editor Pro	2.5.0.1	Analysis of plists of iOS devices
Fiddler	5.0.20192.25091	Network Traffic Capture + Analysis for iOS devices
DB Browser for SQLite	3.11.2	Analysis of SQL databases of Android + iOS devices
Checkra1in	0.9.5	Jailbreak iOS device
Packet Capture	1.7.0	Network Traffic Capture + Analysis for Android devices
MiXplorer	6.42.3	Analysis of rooted tablet
Busybox Free	1.26.2-Stericson	DD image extraction of Galaxy S5 with Netcat implementation

VMDK format on the host machine. Two VMDK files from the emulated device were found under `C:\Program Files (x86)\Microvirt\MEmu\MemuHyperv VMs\MEmu`. Both image files were analyzed using Autopsy and FTK Imager. The first image file contained various system partitions. The second image file contained the device's root partition. Lastly, the Samsung Galaxy tablet and iPhone 6s were jail-broken to gain root access. This will be discussed in the following section.

Logical acquisition for the Galaxy S5 was performed using *dd* with root access. The device was connected to a Windows 10 laptop via USB. *ADB* was used to install *BusyBox* on the device. This allowed for the use of *netcat* as a listener. The *netcat* listener served the output from *dd* over the network. After opening the proper port with *ADB*, a *netcat* client connection was established from the laptop. This connection resulted in the transfer of the image from the Galaxy S5 to the laptop.

All of the tools used in capturing network traffic and acquiring logical images is shown in Table 5.

3.2.1. Gaining root access

While it may be necessary to jailbreak the iOS device or root the Android device in order to gain root access to its system files, analysts should be aware of the implications and consequences. For example, a jailbroken or rooted device would be more susceptible to malware because it removes the restrictions placed by the manufacturer to prevent such attacks. Along the same line, the apps downloaded would have full access to the device, which may result in the device crashing. Also, depending on the tool used to perform the jailbreaking or rooting, system updates could erase the jailbreaking or rooting and the dependent apps; thus, forcing one to go through the process again February.

The Galaxy S5 was rooted using *CF-Auto-Root* in conjunction with *Samsung Odin*. These tools were used from a Windows 10 laptop with the device connected via USB. After the device was rooted, *SuperSU* was installed to manage root privilege access. This allowed for the extension of root privileges to a terminal emulator application, enabling privileged command execution directly on the device. *Android Debug Bridge (ADB)* was also used to gain shell ac-

Table 6
Findings summary.

Artifact types	iOS	Android
Network traffic		
Phone number + Email	√	√
Authentication token	√	√
User's location	√	√
Crossing location w/ match	√	√
Messages	√	√
Audiovisual files	√	√
User's profile info	√	√
Matches' profile info	√	√
Matches	√	√
Proof of user's real info		
Physical Image		
Phone number + Email		
Authentication token		
User's location	√	
Crossing location w/ match		
Messages	√	
Audiovisual files	√	√
User's profile info	√	√
Matches' profile info	√	√
Matches	√	
Proof of user's real info	√	
iTunes Backups		
Phone number + Email		
Authentication token		
User's location	√	
Crossing location w/ match		
Messages	√	
Audiovisual files	√	
User's profile info	√	
Matches' profile info	√	
Matches	√	
Proof of user's real info		

Table 7
Findings + tools used.

Artifact types	iOS	Android
Network traffic	Fiddler	Packet capture
Phone number + Email		
Authentication token		
User's location		
Crossing location w/ match		
Messages		
Audiovisual files		
User's profile info		
Matches' profile info		
Matches		
Proof of user's real info		
Physical image		
Phone number + Email		
Authentication token		
User's location	DB Browser	
Crossing location w/ match		
Messages	DB Browser	
Audiovisual files	DB Browser	
User's profile info	DB Browser, MOBILedit	MiXplorer, DB Browser, Autopsy
Matches' profile info	DB Browser	MiXplorer
Matches	DB Browser	DB Browser
Proof of user's real info	plist Editor Pro, MOBILedit	
iTunes backups		
Phone number + Email		
Authentication token		
User's location	DB Browser	
Crossing location w/ match		
Messages	DB Browser	
Audiovisual files	DB Browser	
User's profile info	DB Browser	
Matches' profile info	DB Browser	
Matches	DB Browser	
Proof of User's real info		

cess to the device from the laptop. Regardless of the method used, the *su* command was used to elevate privileges.

The Galaxy Tab A6 was rooted using *Odin 3.10.7*, *TWRP*, and *SuperSU*. Odin allowed the device to start up from TWRP. To boot with TWRP the user has to press Volume up + Home + Power, although this varies for other devices. TWRP allowed us to flash the SuperSU file which granted root privilege access.

The MEmu emulator came with root access by default. Unlike the Galaxy S5, *SuperSU* was not installed. However, it was still possible to open a shell on the device via *ADB* and run *su* to log in as the root user.

To jailbreak the iPhone 6s, a semi-tethered app called *checkra1n* was used. This works for iPhone 5s – iPhone X and the iOS versions 12.3 and up. At the time of this paper, it was only compatible with Mac OS. Because of this, a Macbook Air laptop was used to install checkra1in through Cydia Impactor. After following the instructions within the app, the iPhone was successfully jail-broken. To gain access to the system files, OpenSSH was utilized. From a Windows 10 laptop, Powershell was run to ssh into the phone by typing in the following command: *ssh root@ip address of phone*. To login, the default password was *alpine*, which then gave successful access into the phone's system files. The password was then changed to prevent potential hacking of the device.

4. Findings

Having captured the traffic between the devices and *Happn* server and analyzing their images, an adequate amount of forensic artifacts was found for both Android and iOS. The artifacts found off the network traffic, logical and physical images for Android will be discussed first. The second section will go over the artifacts for iOS, which includes those from the network traffic, logical and physical images, and the iTunes backups.

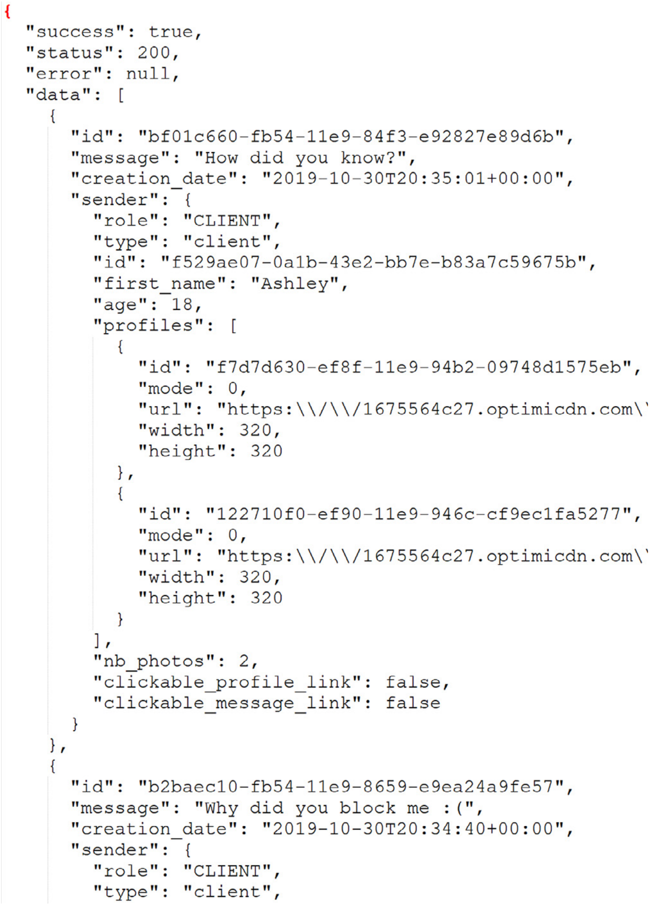
4.1. Android

4.1.1. Network traffic

Network traffic from the Android devices was obtained using the Packet Capture app. Several types of artifacts were located in the captured traffic including text-based messages, profile information, audio messages, and profile pictures.

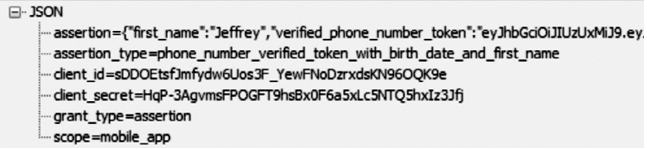
User data was requested through the `/api/users/user_id` endpoint. User information was returned in JSON format. Some of the information included the user's first name, age, school, profile pictures, and distance. The response data also included an entry titled *last_meet_position*. This entry contained latitude and longitude values along with timestamps. One example was observed where the Galaxy S5 (with Trayvon logged in) requested Ashley's data. The GPS coordinates landed just outside of the building where the devices with the two profiles were situated. The response data also contains an entry called *my_relations*. In the same response, the *my_relations* entry included a sub-entry called *mutual* with a timestamp of Ashley and Trayvon's match time.

The client devices requested *Happn* conversation data using HTTP GET requests to the `/api/conversations/conversation_id`. The server sent back JSON responses with the conversation contents and metadata. The response data was filled with separate entries for each message. Each message contained the ID of the sending user, message contents (text, audio, or a Spotify link). Sender information was also provided for each message, including the user's first name, age, and profile pictures. A sample HTTP response with conversation data is seen in Fig. 4.



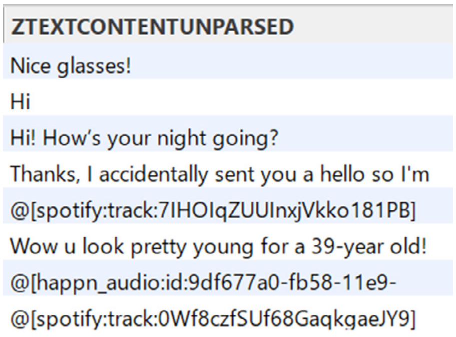
```
{
  "success": true,
  "status": 200,
  "error": null,
  "data": [
    {
      "id": "bf01c660-fb54-11e9-84f3-e92827e89d6b",
      "message": "How did you know?",
      "creation_date": "2019-10-30T20:35:01+00:00",
      "sender": {
        "role": "CLIENT",
        "type": "client",
        "id": "f529ae07-0a1b-43e2-bb7e-b83a7c59675b",
        "first_name": "Ashley",
        "age": 18,
        "profiles": [
          {
            "id": "f7d7d630-ef8f-11e9-94b2-09748d1575eb",
            "mode": 0,
            "url": "https:\\/\\/1675564c27.optimicdn.com\
            "width": 320,
            "height": 320
          },
          {
            "id": "122710f0-ef90-11e9-946c-cf9ec1fa5277",
            "mode": 0,
            "url": "https:\\/\\/1675564c27.optimicdn.com\
            "width": 320,
            "height": 320
          }
        ],
        "nb_photos": 2,
        "clickable_profile_link": false,
        "clickable_message_link": false
      }
    },
    {
      "id": "b2baec10-fb54-11e9-8659-e9ea24a9fe57",
      "message": "Why did you block me :(",
      "creation_date": "2019-10-30T20:34:40+00:00",
      "sender": {
        "role": "CLIENT",
        "type": "client",
```

Fig. 4. Conversation JSON data



```
⊟ JSON
  assertion={"first_name":"Jeffrey","verified_phone_number_token":"eyJhbGciOiJIUzUxMiJ9.ey
  assertion_type=phone_number_verified_token_with_birth_date_and_first_name
  client_id=sDDOEtsfJmfydw6Uos3F_YewFNoDzrxdsKN96OQK9e
  client_secret=HqP-3AgvmsFPOGFT9hsBx0F6a5xLc5NTQ5hxIz3Jfj
  grant_type=assertion
  scope=mobile_app
```

Fig. 5. Jeffrey Lamcee's authentication token.



ZTEXTCONTENTUNPARSED
Nice glasses!
Hi
Hi! How's your night going?
Thanks, I accidentally sent you a hello so I'm
@[spotify:track:7IHOIqZUUInxjVkko181PB]
Wow u look pretty young for a 39-year old!
@[happn_audio:id:9df677a0-fb58-11e9-
@[spotify:track:0Wf8czfSUf68GaqkgaeJY9]

Fig. 6. Messages

Each of the profiles sent audio messages in addition to text. There was no functionality in the app allowing users to download the recordings, only a simple playback feature. However, several of these audio messages were located through the Packet Capture app and successfully downloaded. The *Happn* app pulled the audio messages using HTTP GET requests to an API endpoint with the format `/media/audio/:id`. The ID values were referenced in the JSON responses from `/api/conversations` in the message field.

4.1.2. Physical image artifacts

Within the rooted Android tablet were some interesting artifacts. With MiXplorer, the root file directory was accessed and information was pulled out from both *Happn* and Packet Capture. Under `/data/data/com.ftw_and_co.`*Happn*`/databases` there was information regarding profile data of the user and others. The database file contained text referring to all the users that the Samantha profile crossed paths with, including users that Samantha did not match with. The information includes: age, profile biography, gender, whether the user has sent a like, and URLs that link to pictures of the user. All of this data would allow someone to easily build a profile on a user or track them down. With the images, it would be possible to do a reverse image search and find the user on other social media platforms.

Under `/data/data/com.ftw_and_co.`*Happn*`/shared_prefs/device_attributes.xml`, information about the user's device can be seen. This includes OS type and version number, hardware make and model, *Happn* id, mobile token, and whether Bluetooth is enabled or not. In this case we can tell what device Samantha is using and find ways to exploit her device knowing its specifications. This information could paint a picture for different ways an attacker could access a device.

Happn stores cached photos on the device under `/data/data/com.ftw_and_co_`*Happn*`/cache/http-picasso`. Results were inconsistent among the different devices, but the Galaxy S5 that was used with Trayvon Jackson's profile contained a variety of images. Some of the images were simply application assets, but profile pictures appeared as well. Trayvon's picture was cached along with those of Haley, Ashley, and a real user of the app. media platforms.

4.2. iOS

4.2.1. Network traffic

An abundance of data was found when sniffing the network traffic between the *Happn* app and its server. *Happn* allows a person to sign up or login through either Facebook or a phone number. If a user signs up with a phone number, a verification code is sent to that user via SMS on the provided phone number. The user then enters the verification code into the app to complete the sign up process. There were two packets found in relation to this process. The first is titled */api/verification-sms* and contains the phone number to which the verification code was sent. The other, */api/verification-sms/code* contains the code itself. Additionally, a packet called */connect/oauth/token* was found. This contains the access token for the user along with his/her associated user ID all in plain-text. What was found was that the access token value never changed for each session. As it is used to authorize access and then as a credential Auth0, an attacker could use the token to gain access to the target's account.

The next group of artifacts revolves around users' profiles. A profile on *Happn* mainly consists of preferences that must be filled out, such as partying, cooking, relationship, etc. Other information such as a short biography or occupation are optional. When looking at the captured traffic for the test profile, Jeffrey Lamcee, a packet named */api/users/user_id* gave all profile information on the user, including those mentioned, personal data (age, birthday, registration date), and URLs to his profile pictures. In *api/suggested-users*, the same data was found on profiles that had been rejected and liked, with some having a "my_relationship" value that states if they had been rejected. One significant artifact was coordinates on the last meeting time between Jeffrey Lamcee and the other user(s). These artifacts were repeatedly found in other packets as well, including */api/v1/opportunities* and those that contained a specific user ID. Concerning the test user, his coordinates, city, and country were listed in another iteration of */api/users/user_id*.

Messages had been sent between the pairs of test devices. In the case of Jeffrey Lamcee, he had been matched with Samantha Kasper. Messages that spanned the three sessions were found mainly in */api/conversations/conversation_id* where text, audio, and Spotify song links could be seen, as well as who sent what. There are specific packets dedicated to audio messages or Spotify songs and while the latter contains album and artist information and a playable preview, audio messages could not be played. The two test profiles also blocked each other at certain points. Certain packets with Jeffrey's user ID contained indication of blocking Samantha, including *Message* = *user blocked* and the reason why her account was blocked. When the user was unblocked, another packet contained *Message* = *user unblocked*.

There is a feature in the app that allows a user to explore a map to see in which area paths had been crossed with other "*Happn*ers." To see if network artifacts could be obtained from this, the test accounts moved over large areas of the city and zoomed in on certain regions/counties. While no data on other profiles leaked, what did was that various areas of the city that had been explored on the map could be read. This was seen as significant because it could tell an attacker about the target's geographic preferences and/or general living location.

4.2.2. iTunes backup artifacts

Three iTunes backups from each session were made for each iPhone. The property lists and SQL databases were analyzed and while the former did not have many significant artifacts, the opposite was true for the databases. *Happn* has two main SQL databases to store data: Hdata.db and happSightAnalytics.db. It is the first file that contains user data. There are a number of tables within this database that are of interest, starting with ZFLHPDEVICE and ZFLHPRECOVERYINFORMATION. The first table contains coordinates of the user's location at the time of app use, country, language, and city while the second has the user's email address, which *Happn* uses to log a user in when an issue is run into. ZFLHPIMAGE contains URLS to other users' profile pictures, but also has Z_PK values that when matched with a PROFILEPICTURE value in the ZFLHPPROFILEITEM table, will connect user's name to his/her photos(s). Continuing, ZFLHPMEDIA shows the sent audio messages that while including URLs, were still not playable. Messages between the test accounts are found in ZFLHPMESSAGE where text, audio, and Spotify messages can be seen. The table also has a ZSENDER value that is associated with a certain user – which is consistent throughout the tables and sessions – and a ZDATESENT value that when decoded tells the date the message was sent. As mentioned before, the ZFLHPPROFILEITEM table can be used to connect user to profile image. What it also contains is various values about the potential profiles a user rejects or likes, including first and last name, job, and workplace. It should be noted however that some of these values were NULL for an unknown reason that is not associated with whether a profile was rejected or liked. Additionally, ZFLHPUSERREJECTED has a list of those users who were rejected. The profile preferences of the test user were found in ZFLHPNTRAITSINGLEOBJECTANSWER and lastly, ZFLHPREPORT shows when a profile is blocked with that user's associated value and the reason for blocking.

4.2.3. Physical image artifacts

Once root access was gained in the iPhone 6s, some additional artifacts could be found. An interesting one relates to the Hdata.db. After the last session, the account was "paused" to see if any artifacts could still be retrieved. When looking at the database afterwards, it appeared empty and only began storing data when the account was logged back on and activity was conducted. Another interesting artifact is one that could indicate if an individual created a false profile. The device name was found in

/private/var/root/Library/Lockdown/data_ark.plist, a property list that has data on the device and account holder. In the case of Jeffrey Lamcee, which was depicted as a false profile, the *DeviceName* value was "John's iPhone." If the name of the device holder is different from the app profile on it, that could indicate the creation of a fake online profile.

Other artifacts include the following. Jeffrey Lamcee's two profile pictures could be downloaded and viewed from /private/var/mobile/Media/DCIM/100APPLE. One log named *Mobile_log.0* states in which container the application bundle (in the case of *Happn*, it is fr.ftw-and-co.whoozer) gets stored, allowing one with access to pinpoint that exact directory and obtain the app's data. *Com.apple.springboard.plist* tells which apps are on a device's home screen and on the iPhone 6s, *Happn*'s bundle name was found as one of the apps. Lastly, /private/var/mobile/Library/CoreDuet/Knowledge/ contains a *KnowledgeC.db* database that stores data on multiple processes that run in an iOS device, including application usage (How a Suspect?s Pattern-of-life Analysis is Enhanced with KnowledgeC Data, 2019). When looking at one of its tables, ZOBJECT, app usage can be viewed in correspondence with a certain application, its stream name, UUID, and start date, among other values. Therefore, it could be seen that fr.ftw-and-co.whoozer was used on December 6, the day Jeffrey Lamcee's account was unpaused.

5. Limitations

During the experiment, there were some attributes of the app and issues with the tools used that limit the findings. For iOS, there were two. One involved the *Happn* app update to 9.7.0. It was found that with the update one phone number could not be used for different accounts. This was not an issue before because two of the test devices had been signed up with the same phone number. If a used phone number is associated with a new account, *Happn* will override the original account with the newly-created one. Despite the new implementation providing a more secure form of authentication, the test profiles containing the same phone number – Ashley Gray and Hayley Thomas – had to be recreated after having created forensic artifacts. The second issue was that SQL database data from November 7 for two of the profiles – Jeffrey Lamcee and Dottie Harford – contained no data, despite there being data for the same app database for the dates, October 30, 2019, and November 6, 2019. The reason for this is because the November 7, 2019, session involved "pausing" the accounts, essentially logging the users out. When this occurs, *Happn* does not store the previous data in its database. The tablet device containing Samantha Kasper's information was unable to be used in conjunction with Autopsy. The program was unable to obtain information from the root directory. To solve this issue, we used MiXplorer to view said files.

As for the Android devices, three limitations were found. The first is that a SIM card is not needed to login the app with a phone number. For an iOS device, a user could not sign up for a *Happn* account with his/her phone number unless a SIM card was present. This was not an obstacle for the tested Android device. The second issue is that one phone number can be tied to different accounts. This can be seen as an authentication hole through which a user can set up various profiles based on a single phone number. Lastly, during the capturing of the Android devices' network traffic, the Packet Capture app could not work when running on the Galaxy S6. Because of this, the associated profile's – Joseph Jordan – app activity could not be captured.

It should also be highlighted that the image acquisition tool, MOBILedit Forensic Express, did not acquire app or system data that hadn't already been found by the other tools for both iOS and Android devices. This could be due to the fact that its application analysis feature does not currently support the *Happn* app and therefore could only retrieve a limited amount of information.

6. Conclusion and future research

In this study, mobile device forensics and network analysis on the *Happn* dating app were conducted. The goal was to identify artifacts that may pertain to sensitive user data, using both iOS and Android devices along with eight (8) varying user profiles with diverse backgrounds throughout the investigation. Each device ran different versions of the iOS and Android *Happn* dating app. The information we were able to recover was found through the use of many tools such as FTK Imager and Packet Capture. What was found was self-revealing information of the user and profiles they interacted with, information such as exact location with longitude and latitude, URL links containing all photos uploaded by the user and others they interacted with, access tokens, user Ids, and profile information posted by users.

On a broader note, the mobile forensics 'space' can be seen as a race, not only to keep up with device (i.e. hardware) and software releases by providers (e.g. new app version or addition of new features / functions), but also from software and hardware modifications made by end users, particularly serious and organized criminals, to complicate or prevent the collection and analysis of digital evidence. Hence, future mobile forensic tools and/or approaches should be designed in such a way that it can be readily adapted to the latest mobile device and app technologies. This is essential given the ongoing and rapid change characteristic of mobile computing technology and to ensure the digital forensic practitioner and researcher communities have timely expertise and insights to their potential use in criminal activities, particularly in serious and organized criminal activities. Thus, it is important to keep a watchful brief on the threat landscape, such as the following:

1. How are dating apps, and more generally mobile apps used in the commission and execution of crimes and incidents that require digital investigations?
2. What types of evidential data (e.g. data-at-rest and data-in-transit) are available, considering the advanced security features and anti-forensic techniques that could be utilized, particularly by serious and organized criminals?
3. What techniques can the digital forensic investigators use to legally gain access to the identified evidential data?

Future agenda also include extending this research to other popular dating apps, and presenting a categorization of potential forensic artifacts that could be recovered from the analysis of these popular dating apps. This will allow the forensic community to know what forensic artifacts they should be looking for in the examination of such dating apps. In addition, the findings from the categorization can also help inform the design of future mobile forensic tools, for example to circumvent future security measures without compromising the evidence's integrity.

It is also important to balance the need for a secure mobile / telecommunications system and the rights of individuals to privacy against the need to protect the community from criminal activities, particularly serious and organized crimes, and cyber security interests. This is an issue that has serious implications on the ability of governments to protect their citizens. However, it remains an under-researched area due to the interdisciplinary challenges specific to this research. Hence, there is also a need for mobile app developers to place a greater focus on the right to individual privacy, for example by following existing industry best practices for designing Secure mobile apps[3].

[3] https://developer.android.com/topic/security/best-practices, last accessed March 31, 2020.

Acknowledgements

The authors thank MOBILedit for providing us free educational MOBILedit Forensic Express licenses. A number of commercial entities, equipment, or materials have also been identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the authors and their employers, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

References

Agrawal, Animesh Kumar, Khatri, Pallavi, Sinha, Sumitra Ranjan, 2018. Comparative study of mobile forensic tools. Advances in Data and Information Sciences. Springer, pp. 39–47.

Anglano, Cosimo, Canonico, Massimo, Guazzone, Marco, 2020. The Android Forensics Automator (AnForA): A tool for the Automated Forensic Analysis of Android Applications. Computers & Security 88 101650.

Atkinson, J.S., Mitchell, J.E., Rio, M., Matich, G., 2018. Your WiFi is leaking: What do your mobile apps gossip about you? Future Generation Computer Systems 80, 546–557. doi:10.1016/j.future.2016.05.030. URL http://www.sciencedirect.com/science/article/pii/S0167739X16301480.

Attrill-Smith, Alison Lloyd Joanne, Chris, Fullwood, 2019. Online romantic relationships. The Oxford Handbook of Cyberpsychology, 195.

Barmpatsalou, Konstantia, Cruz, Tiago, Monteiro, Edmundo, Simoes, Paulo, 2018. Current and future trends in mobile device forensics: A survey. ACM Computing Surveys 51 (3), 46.

Cahyani, Niken Dwi Wahyu, Choo, Kim-Kwang Raymond, Ab Rahman, Nurul Hidayah, Ashman, Helen, 2019. An evidence-based forensic taxonomy of windows phone dating apps. Journal of Forensic Sciences 64 (1), 243–253.

Di Luzio, A., Mei, A., Stefa, J., 2018. Uncovering hidden social relationships through location-based services: The Happn case study. IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 802–807. doi:10.1109/INFCOMW.2018.8406866. ISSN: null.

February, L. R., 2017. 10 Pros and Cons of Jailbreaking Your iPhone or iPad. Library Catalog: www.tomsguide.com, URL https://www.tomsguide.com/us/pictures-story/.

Gurugubelli, Dheeraj, Gino, Lourdes, Rogers, Marcus K., 2015. What lies beneath?: the forensics of online dating. In: Proceedings of the 16th Annual Information Security Symposium. CERIAS-Purdue University, p. 32.

How a Suspect?s Pattern-of-life Analysis is Enhanced with KnowledgeC Data, 2019. URL: https://www.cellebrite.com/en/blog/how-a-suspects-pattern-of-life-analysis-is-enhanced-with-knowledgec-data/

Jadhav Bhatt, A., Gupta, C., Mittal, S., 2018. Network Forensics Analysis of iOS Social Networking and Messaging Apps. 2018 Eleventh International Conference on Contemporary Computing (IC3), pp. 1–6. doi:10.1109/IC3.2018.8530576. ISSN: 2572-6110.

Kim, Kuyju, Kim, Taeyun, Lee, Seungjin, Kim, Soolin, Kim, Hyoungshick, 2018. When Harry Met Tinder: Security Analysis of Dating Apps on Android. In: Gruschka, N. (Ed.), Secure IT Systems. Springer International Publishing, pp. 454–467.

Kim, Dohyun, Lee, Sangjin, 2020. Study of identifying and managing the potential evidence for effective Android forensics. Forensic Science International: Digital Investigation.

Lauckner, Carolyn, Truszczynski, Natalia, Lambert, Danielle, Kottamasu, Varsha, Meherally, Saher, Schipani-McLaughlin, Anne Marie, Taylor, Erica, Hansen, Nathan, 2019. "Catfishing," cyberbullying, and coercion: An exploration of the risks associated with dating app use among rural sexual minority males. Journal of Gay & Lesbian Mental Health 23 (3), 289–306.

Lee, M., 2018. Crime and the cyber periphery: Criminological theory beyond time and space. The Palgrave Handbook of Criminology and the Global South. Springer, pp. 223–244.

Mata, Nicholas, Beebe, Nicole, Choo, Kim-Kwang Raymond, 2018. Are Your Neighbors Swingers or Kinksters? Feeld App Forensic Analysis. 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, pp. 1433–1439.

Murphy, Alyssa, 2018. Dating dangerously: Risks lurking within mobile dating apps. Catholic University Journal of Law and Technology 26 (1), 7.

Scannell, Meredith Jean, 2019. Online dating and the risk of sexual assault to college students. Building Healthy Academic Communities Journal 3 (1), 34–43.

Schreurs, Lara, Sumter, Sindy R., Vandenbosch, Laura, 2020. A Prototype Willingness Approach to the Relation Between Geo-social Dating Apps and Willingness to Sext with Dating App Matches. Archives of Sexual Behavior 1–13.

Shetty, Rushank, Grispos, George, Choo, Kim-Kwang Raymond, 2020. Are you dating danger? An interdisciplinary approach to evaluating the (in) security of Android dating apps. IEEE Transactions on Sustainable Computing.

Sudozai, M.A.K., Saleem, S., Buchanan, W.J., Habib, N., Zia, H., 2018. Forensics study of IMO call and chat app. Digital Investigation 25, 5–23. doi:10.1016/j.diin.2018.04.006. URL: http://www.sciencedirect.com/science/article/pii/S1742287618300094.

Sumter, Sindy R., Vandenbosch, Laura, 2019. Dating gone mobile: Demographic and personality-based correlates of using smartphone-based dating applications among emerging adults. New Media & Society 21 (3), 655–673.

Tomaszewska, P., Schuster, I., 2019. Comparing sexuality-related cognitions, sexual behavior, and acceptance of sexual coercion in dating app users and non-users. Sexuality Research and Social Policy 1–11.

Veel, K., Thylstrup, N.B., 2018. Geolocating the stranger: the mapping of uncertainty as a configuration of matching and warranting techniques in dating apps. Journal of Aesthetics & Culture 10 (3), 43–52.

Yadegarfard, M., 2019. How are Iranian gay men coping with systematic suppression under islamic law? a qualitative study. Sexuality & Culture 1–24.

Zhang, Xiaolu, Choo, Kim-Kwang Raymond, 2020. Digital Forensic Education: An Experiential Learning Approach. Studies in Big Data , 61. Springer.

Shawn Knox received his Bachelor's of Business Administration in Cyber Security from UTSA in 2019. In the same year, he has begun pursuing a Master's of Science in Information Technology with a concentration in Cyber Security. His goals are to expand his knowledge of the cyber security field and apply that knowledge as a security analyst for the big businesses.

Steven Moghadam received his Bachelor's of Business Administration in both Information Systems and Cyber Security from UTSA in Fall 2019 and has since been pursuing a Master's of Science in Information Technology with a concentration in Cyber Security. He has been awarded the Department of Defense (DoD) Cyber Scholarship Program (CySP) as part of his Graduate degree, which allows him to further his studies through the master's cyber program. His career goal consists of working as a Cyber Security Specialist for the US Government.

Kenny Patrick received his Bachelor's of Business Administration in Cyber Security from UTSA in December 2018. He began pursuing a Master's of Science in Information Technology at UTSA in January 2019 with an expected graduation in May 2020. His interests include digital forensics, incident response, software development, and automation.

Anh Kim Phan received her Bachelor's of Business Administration in Cyber Security from UTSA in 2018 and has since been pursuing a Master's of Science in Information Technology with a concentration in Cyber Security. She has been awarded the Kudla Endowed Fellowship in Information Assurance and Security as part of her Graduate degree, which allows her to work as a Research Assistant on multiple research projects in the field of Cyber Security. Her career goal consists of working as a Digital Forensic Analyst for the US Government.

Kim-Kwang Raymond Choo holds the Cloud Technology Endowed Professorship at UTSA. In 2016, he was named the Cybersecurity Educator of the Year - APAC, and in 2015 he and his team won the Digital Forensics Research Challenge organized by Germany's University of Erlangen-Nuremberg. He is the recipient of the 2019 IEEE TC on Scalable Computing's Award for Excellence in Scalable Computing (Middle Career Researcher), 2018 UTSA College of Business Endowed Research Award for Tenured Faculty, 2018 IEEE Access Outstanding Associate Editor, British Computer Society's 2019 Wilkes Award Runner-up, 2019 EURASIP JWCN Best Paper Award, Korea Information Processing Society's JIPS Survey Paper Award (Gold) 2019, IEEE Blockchain 2019 Outstanding Paper Award, IEEE TrustCom 2018 Best Paper Award, ESORICS 2015 Best Research Paper Award, 2014 Highly Commended Award by the Australia New Zealand Policing Advisory Agency, Fulbright Scholarship in 2009, 2008 Australia Day Achievement Medallion, and British Computer Society's 2018 Wilkes Award.