A FRAMEWORK FOR DESCRIBING MULTIMEDIA CIRCULATION IN A SMARTPHONE ECOSYSTEM

Panagiotis Andriotis, Theo Tryfonas, George Oikonomou, Irwin King

Chapter 15

A FRAMEWORK FOR DESCRIBING MULTIMEDIA CIRCULATION IN A SMARTPHONE ECOSYSTEM

Panagiotis Andriotis, Theo Tryfonas, George Oikonomou and Irwin King

Abstract Contemporary mobile devices allow almost unrestricted sharing of multimedia and other types of files. However, because smartphones and tablets can easily access the Internet and exchange files wirelessly, they have also become useful tools for criminals who perform illegal activities such as sharing contraband and distributing illegal images. Thus, the need to investigate the source and destination of a multimedia file that resides in the internal memory of a smartphone is apparent. This chapter presents a framework for illustrating and visualizing the flow of digital images extracted from Android smartphones during a forensic investigation. The approach uses "big data" concepts to facilitate the processing of diverse (semi-structured) evidence from mobile devices and extends the idea of digital evidence bags. The data used for evaluation was obtained by running experiments that involved image exchange through channels such as Bluetooth, Internet and cloud services. The study presents information about the locations where evidence resides and uses graph databases to store metadata and to visualize the relationships that connect images with apps and events.

Keywords: Android forensics, graph database, content, analysis, NoSQL

1. Introduction

The proliferation of smartphones and fast mobile telephony networks offer countless opportunities for users to exchange text messages, photographic images, videos and multimedia content. Unfortunately these smart applications, which are equipped with convenient interfaces that facilitate the smooth and rapid flow of information, have become tools

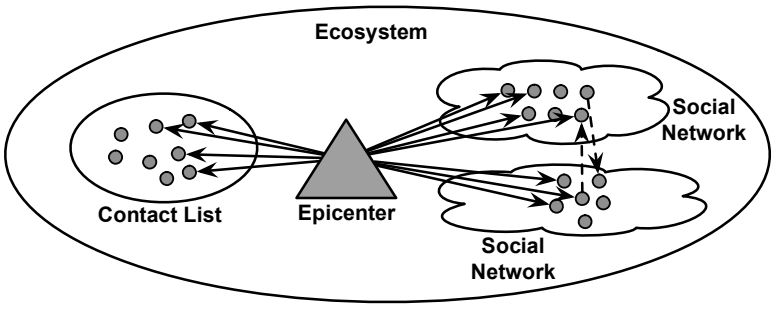


Figure 1. Smartphone ecosystem.

in the hands of criminals who commit crimes such as sharing contraband and distributing illegal images.

Smartphones are often seized in investigations and the evidence extracted from them is used in courts to establish the guilt or innocence of defendants. They are valuable evidence containers because they hold large amounts of personal information and are fully integrated with their owners' lifestyles. Smartphones are equipped with a variety of sensors that enable developers to create powerful applications (apps). They have also become very efficient with regard to their processing power and the accuracy of their sensors. Users are able to connect to the Internet, upload and download material through cloud services, capture images, sound recordings and videos, monitor their health and well-being, create and edit documents and spreadsheets, and obtain personalized information from various sources based on their locations and interests. All these actions leave artifacts that can be used as evidence.

A smartphone can be viewed as defining a unique "ecosystem" where the owner of the device is a central entity: the epicenter. The ecosystem consists of smaller groups that include entities linked with each other via various types of relationships. For example, the contact list of a smartphone corresponds to an entity group in which an analyst can find people who are connected with the epicenter. Another entity group might include the accounts of friends in a social network or the contacts in a professional network like LinkedIn. One of the problems that a forensic analyst has to solve is whether an entity is a real person or whether it represents a fake, secondary or parody account. The forensic analysis of a smartphone can also reveal entities from different groups that are linked together (especially, if the smartphone owner has used the contact syncing utility provided by many social network apps).

Figure 1 shows a smartphone ecosystem. The epicenter is linked with the contact list group (neighborhood) and with two other social networks. Some entities in the ecosystem are also linked together (dashed

lines) via the automated contact linking process. During a forensic analysis, the different neighborhoods can be linked via the artifacts found in various databases in the internal memory of the smartphone. In Android smartphones, applications store data in specific folders in the data partition [8]. Most of the application folders contain the folders *databases*, *cache*, *lib* and *files*. The *databases* folder is usually where information about the user is maintained in SQLite databases. The stored information can help reconstruct the profile and activities of the epicenter. Furthermore, a forensic analysis can be enhanced using reporting tools that provide visual metaphors of the underlying data [18].

This approach involves separating diverse data existing in the smartphone internal memory into distinct categories such as entities, groups and multimedia. It also focuses on the relationships with the epicenter. However, despite the practicality of the representations of social connections, certain disadvantages exist. The approach does not provide additional information about the events linked to entities. For example, if a forensic analyst wishes to visualize the entity in the ecosystem who is responsible for capturing and distributing an illegal image, Figure 1 would not provide any useful hints because the entities are not linked with actions they performed using the various apps. Consequently, it is necessary to incorporate functionality that links actions with entities.

This chapter focuses on cases involving the circulation of digital images (photographs) in a smartphone ecosystem. The goal is to construct a network that depicts the distribution of multimedia in the ecosystem defined by an Android mobile device. The main contribution is a framework that solves the problem of linking entities with events and digital artifacts during forensic analyses of smartphones while highlighting the relationships between apps and multimedia. The approach uses big data concepts and extends the concept of digital evidence bags, utilizing modern storage methods and focusing on the ecosystem epicenter and its actions. Information stored about an individual under investigation is augmented with visualizations of the interactions in his ecosystem using graph databases. The conceptual design is readily extended to cover other types and aspects of evidence found in smartphones. In addition, the framework can represent multiple smartphone ecosystems that are linked in a case.

2. Related Work

Several researchers have highlighted the importance of identifying traces that reveal access to illegal content. For example, Howard [9] has presented a technical analysis of the cache and the methods that a

forensic analyst may use in a child pornography case to find data stored in the *Temporary Internet Files* folder of a browser.

More recently, the seamless information flow provided by peer-to-peer (P2P) networks has transformed digital neighborhoods to blooming areas of illegal image and video trafficking. Hurley et al. [11] have analyzed multimedia trafficking in two popular P2P networks (eMule and Gnutella) and have studied various subgroups such as peers that use Tor, peers that bridge multiple P2P networks and peers that contribute to file availability. They conclude that these groups are more active than others with regard to child pornography trafficking. Wolak et al. [20] have examined data from Gnutella using the RoundUp investigative tool [13] and propose that data should be systematically gathered and analyzed to prioritize investigations in P2P networks. However, these tools cannot be applied to smartphones and used to reconstruct and present the exchange of information between different apps.

Traditional digital image forensics [5] can be employed to perform tasks such as device identification and linking, and digital forgery detection. Despite the plethora of anti-forensic software [17], certain information sources exist that can provide indications (e.g., sensor data) that can be used during forensic analyses of smartphones [16]. However, this data is usually volatile and is not available during a post-mortem analysis.

Several novel approaches have been proposed to automatically collect and analyze non-volatile data from Android devices, especially devices used in sensitive enterprise environments [7]. Marturana and Tacconi [15] have proposed a machine-learning-based triage scheme to automate digital media categorization, essentially blending digital forensics and machine learning. Liu et al. [14] have used support vector machines to identify the smartphone camera sources of digital images and to reveal operations that may have been applied to the digital images. Turner [19] has presented an approach for unifying digital evidence from disparate sources using digital evidence bags (i.e., universal containers for capturing and processing digital evidence from different sources). Flaglien et al. [4] have highlighted the obstacles posed by diverse file formats with regard to digital evidence processing; several solutions have been proposed to overcome these obstacles. For example, Garfinkel [6] developed the DFXML scheme to facilitate the exchange of structured forensic data from multiple sources.

Digital forensic researchers have studied the diverse data that can be drawn from the disparate sources existing in a smartphone ecosystem. Chung et al. [3] have analyzed artifacts from various sources (desktop machines and mobile devices) connected to cloud storage services, re-

vealing traces of activity in file paths, XML files and databases. Huber et al. [10] have collected and categorized information from social media networks, including user data, posts, private messages, photograph images and associated metadata; Kontaxis et al. [12] have used this information to detect if cloned profiles exist in social networks. Anglano [2] has analyzed the traces that remain from the use of the popular chatting app, WhatsApp Messenger, on Android devices. However, no work has focused on the circulation of images (or multimedia) in smartphone ecosystems. The framework presented in this chapter integrates all the valuable data existing in a smartphone ecosystem, facilitating flexible and extensible evidence visualization and analysis.

3. Using Graph Databases

Forensic investigations of smartphones involve four basic steps: (i) smartphone seizure; (ii) logical and/or physical data acquisition; (iii) data analysis; and (iv) data presentation and preservation. This study focuses on the data presentation step and proposes a methodology for automating forensic investigations of illegal image trafficking in smartphone ecosystems. Of particular interest are images that have entered or exited an ecosystem via five paths: (i) wireless technologies such as Bluetooth, Wi-Fi Direct and near field communications (NFC); (ii) email; (iii) Internet downloads; (iv) cloud storage services; and (v) smartphone applications such as Facebook Messenger and Twitter.

The mapping approach highlights the relationships between photographic images and their sources or destinations. For example, the most obvious relationship between a JPEG image and a Camera application is if the photographic image was captured using the app or if it was downloaded or distributed using another app. This information is critical to determining if an individual under investigation was involved in producing illegal content or merely distributing the content.

The storage capacity of modern mobile devices is constantly increasing and forensic analyses of mobile devices involve the processing of very large amounts of information. As a result, a medium such as a database is required to safely migrate and store all the acquired data and its relationships for further analysis. Traditional databases (e.g., relational databases) require very complicated design strategies to depict relations, foreign keys, joins and tables as reference points to link different entities. However, new types of databases have been introduced for social media and big data. These include NoSQL and graph databases such as Cassandra, MongoDB and Neo4j. The principal advantage of a graph

database over a relational database is its ability to easily handle and depict relationships between linked entities.

This chapter investigates relations that occur between linked entities. For example, to locate images that were downloaded from the Internet, connections and/or relationships with the attribute DOWNLOADED must be targeted. Thus, a graph database is the most appropriate storage medium because it supports searches for patterns within paths created by nodes that are linked together. In a graph database, nodes can be stored and connected together using different attributes based on the types of relationships existing between the nodes. The result is an expandable graph that can store different kinds of information. Indeed, photographic images, videos, music and sound clips, and documents can all be stored in a graph database without the need for database refactoring.

4. Use Case Experiments

A scenario involving numerous user activities was designed in order to capture the traces left in smartphone internal memory after the use of several common applications. The smartphone used was a Samsung Galaxy Fame GT-6810P running Android Operating System (version 4.1) and equipped with an external secure digital (SD) card and superuser privileges (su).

The following actions were performed to simulate illegal image exchange and trafficking in the smartphone ecosystem:

- Images were emailed to the user's account. The user viewed and downloaded them to the smartphone. The standard Android Email and Gmail apps were used.
- Images were captured using the smartphone camera and stored on the smartphone.
- Images were exchanged using wireless interfaces (Bluetooth and NFC).
- Applications such as Snapchat (chatting while exchanging images) were used. Other apps included the popular Facebook Messenger, WhatsApp and Google Hangouts.
- Images were uploaded and downloaded from the smartphone via cloud services (Dropbox and Google Drive apps).
- Images were downloaded to the smartphone using applications such as Twitter and Instagram.

- Images were downloaded from the Internet using the standard Android Internet browser and Google Chrome, which is shipped with most recent versions of the Android operating system.

After the actions were performed, the data was collected via a physical acquisition of the smartphone data partition. During this process, the USB Debugging option on the smartphone was enabled and the Android Debug Bridge (`adb`) tool was utilized. As described in [1], this is a common way to extract physical images of smartphone partitions. Physical data acquisition also includes the extraction of deleted images using open source tools such as `scalpel`. Deleted images can be incorporated in the graph database as nodes connected to the epicenter with relationships marked DELETED.

5. Results

Table 1 presents the locations (in the form of lists of folders) where information related to the circulation of images in the ecosystem is stored. It also describes the traces found in the internal memory of a smartphone. Of special interest is the data folder (`data/data/`) of a smartphone where most applications store significant amounts of information. Note also that the databases listed in this section are not encrypted and an individual with superuser privileges can access and view them in an open source SQLite browser such as `sqliteman`. Note that [SQLite] in Table 1 denotes a SQLite3 database. Anglano [2] provides information about decoding `.nomedia` image files.

The Android operating system is installed on a variety of devices with different characteristics. A smartphone, for example, may be equipped with external disk storage (an SD or microSD card), internal emulated storage or both. The test smartphone had two storage folders: emulated and external SD. When an SD card is inserted in a smartphone, images captured by the Camera app are usually stored on the SD card. Logical copies of the folders in the external and emulated media reveal only the images that can be seen by the operating system. However, these images may contain important information in their Exif metadata headers, such as user location if the GPS was enabled when the image was captured. A logical copy of the storage media may be obtained by connecting the smartphone to a computer via a USB cable and issuing a pull command to the `adb` tool. Collecting this data augments a visualization with information about the types of connections that link nodes in the graph. Table 2 lists the possible locations of digital images in the test smartphone.

Table 1. Resources that provide information about image sharing.

Path	Type	Details
com.android.bluetooth/ /databases/btopp.db [SQLite]	Sent, received, deleted	In table btopp, see uri and direction
com.android.email/ /cache/[folder_e.g._1.db_att]	Received, attached (via Email app)	May not be visible via Gallery app
com.android.email/databases/ /EmailProvider.db [SQLite]	Sent, received, downloaded	In attachment and message tables
com.android.providers.downloads/ /databases/downloads.db [SQLite]	Downloaded	Downloaded (Internet) chat apps (e.g., Hangouts)
com.dropbox.android/ databases/db.db [SQLite]	Deleted, uploaded	Tables dropbox and photos
com.facebook.orca/cache/fb-temp/	Uploaded	Uploaded content
com.facebook.orca/cache/image/ /v2.ols100.1/[folders]/	Sent, received, seen (via Gallery app)	Thumbnails from Gallery (if accessed by Messenger)
com.google.android.apps.docs/cache/ /diskCache/fetching/account_cache_1/	Uploaded, downloaded	Images residing in Google Docs app
com.google.android.apps.docs/ /databases/DocList.db [SQLite]	Deleted, data owners	Tables like entry111
com.google.android.apps.docs/ /files/fileinternal/[folders]/	Downloaded, pinned	Pinned images to be viewed offline
com.google.android.gm/ /cache/[user's_gmail_address]	Unknown, sent, received	Images and other attachments
com.google.android.talk/ /cache/scratch/	Sent	Sent images via Hangouts app.
com.google.android.talk/ /databases/babel1.db [SQLite]	Sent, received	In messages, see attribute local_uri
com.instagram.android/cache/	Pending, captured, sent, received	Names flagged with timestamps
com.sec.android.gallery3d /databases/picasa.db [SQLite]	Various	Table photos (if auto-uploading is on)
com.sec.android.providers.downloads/ /databases/sisodownloads.db [SQLite]	Downloaded	Downloaded via Internet browser
com.snapchat.android/cache/ /received_image_snaps/	Received (.nomedia files)	May be encrypted (version dependent)
com.snapchat.android/cache/ /stories/received/thumbnail/	Received (.nomedia files)	May be encrypted (version dependent)
com.snapchat.android/ /databases/tcspahn.db	Various	Entries about sent, received images

Table 2. Locations of images in external and emulated storage.

Path	Type
`mnt/extSdCard/DCIM/Camera/`	Photos captured via the Camera app
`mnt/sdcard0/Download/`	Downloaded via the Chrome browser
`mnt/extSdCard/Download/`	Downloaded via the Internet browser
`mnt/sdcard0/Pictures/Twitter/`	Uploaded and downloaded via Twitter
`mnt/sdcard0/Pictures/Facebook/`	Captured and uploaded via Facebook
`mnt/sdcard0/Pictures/Messenger/`	Downloaded via Messenger
`mnt/sdcard0/Beam/` or `/sdcard0/Bluetooth/`	Exchanged via wireless
`mnt/sdcard0/Snapchat/`	Downloaded photos via Snapchat
`mnt/sdcard0/WhatsApp/Media/ /WhatsApp Images/`	Received and sent images via WhatsApp
`mnt/sdcard0/Android/data/`	Folders containing photos from distinct apps

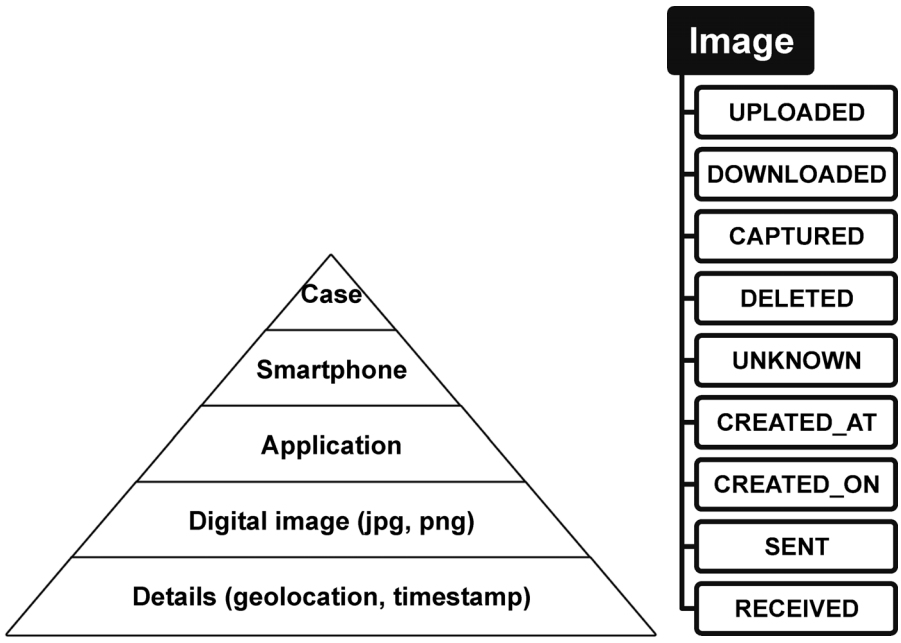
6. System Design

A graph database is a graph and a database at the same time. Entities, photographic images and applications are represented as nodes in the graph. This feature makes the entire system expandable and able to hold diverse types of information (e.g., videos, sound recordings and documents). In the proposed framework, each case is represented as a node in the graph (because, in the future, different cases with unique ecosystems may have to be linked). For example, a person under investigation might be involved in multiple cases. In the graph database, nodes are connected with relationships such as DOWNLOADED, UPLOADED and DELETED, and contain information from the original SQLite databases. Nodes and relationships both contain attributes/properties that can be used by a forensic analyst to find patterns and paths inside the graph by issuing SQL-like queries.

The work described in this chapter used the Neo4j graph database (available from `www.neo4j.org`) and its Cypher query language. Cypher is a graph database language that uses ASCII art expressions and a limited number of commands to issue graph queries. An example of the ASCII art writing style used by Cypher is:

$$(a)->[:UPLOADED]-(b)-[:DELETED]->(c)$$

where (a), (b) and (c) are nodes and [:UPLOADED] and [:DELETED] are relationships. Thus, (a) is linked with (b) via the [:UPLOADED]



(a) Hierarchical node structure. (b) Relationships.

Figure 2. Graph database conceptual design.

relationship that points from (a) to (b), and so on. In a large graph, the query output can be presented graphically, viewed using a browser or outputted as a table. Graph databases are very useful in forensic analysis because they provide the opportunity to visualize data and simultaneously support searches for patterns and paths in the ecosystem using a common infrastructure (which is the graph database itself).

Figure 2(a) presents the hierarchical design of a system that incorporates the findings of the case study discussed in Section 5. The graph has nodes that represent: (i) the case; (ii) the seized smartphone (linked with the case); (iii) the applications existing in the smartphone ecosystem; (iv) the images found in the cache and other storage media; and (v) important details such as geolocations and timestamps. It was decided to depict geolocations and timestamps as distinct nodes and not as node attributes due to the importance of this information and also because it may be necessary to link more actions in the future with specific geolocations and timestamps. Figure 2(b) shows the main relationships that link images with other graph entities. CAPTURED is related to the Exif metadata, which identifies the camera type (or smartphone) used

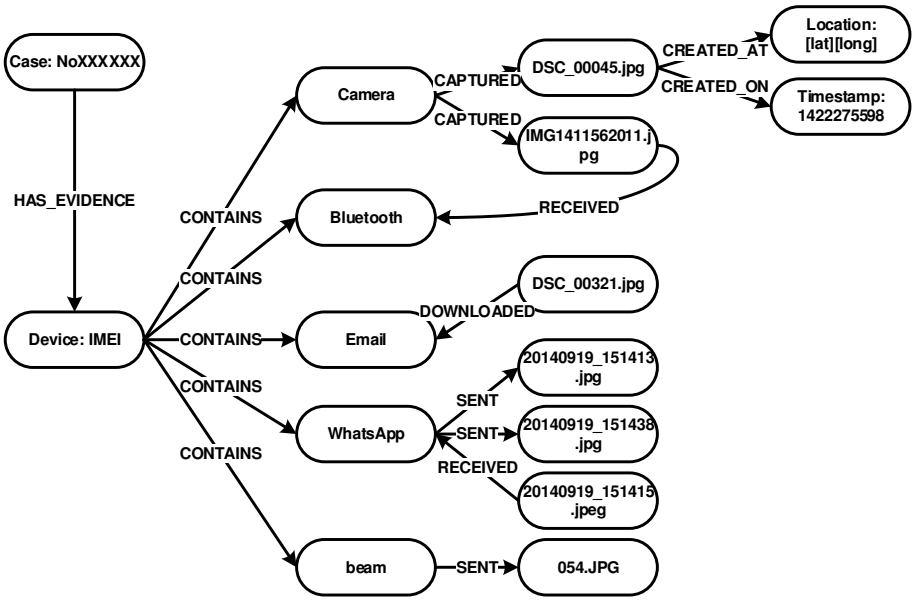


Figure 3. Nodes and their relationships in the graph database.

to capture the image. CREATED_AT is a relationship that links images and locations and CREATED_ON links images with timestamps. Figure 3 shows an example of the principal entities and their relationships in the graph database.

One of the advantages of this system is that it is easily extended to incorporate new nodes corresponding to videos, sound recordings and documents. In addition, it allows for the integration with other ecosystems that might exist in a case. It is also possible to add nodes that represent entities such as social media accounts and individuals in a contact list. Thus, data and evidence found in different devices involved in a case can be linked.

Another advantage of the proposed framework is its graph representation, which makes it possible to apply graph-theoretic approaches and metrics to extract information. Example metrics are the degree, indegree and outdegree of a node corresponding to an application (e.g., Facebook Messenger) that describe the usage frequency of the app and the incoming and outgoing digital evidence, respectively. At a larger scale, these metrics provide information about the most preferred application for image exchange.

The timestamp nodes help focus on different periods of time and highlight potential changes in user behavior. Additionally, data from Neo4j graph databases are easily stored as GraphML files. GraphML files can

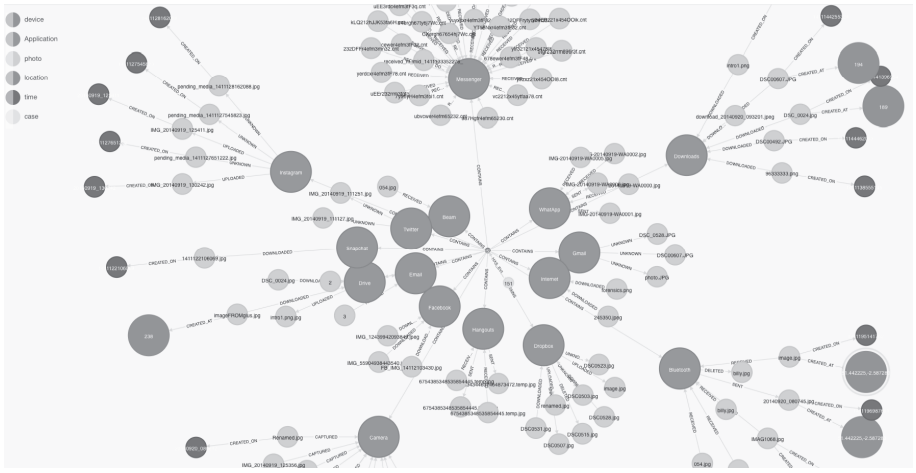


Figure 4. Screenshot from the experimental graph database.

be fed to sophisticated graph visualization tools to significantly enhance data analysis.

Figure 4 shows a screenshot of the graph database after data collection. To view the graph, it is necessary to establish a connection with `http://localhost:7474/browser` when the database server is running and type the command MATCH (n) RETURN n.

Figure 5 shows an example of further analysis of the graph using the Gephi open source visualization tool (available from `gephi.github.io`). The "busy" nodes (i.e., the apps most used for image sharing) were identified by extracting the graph as a GraphML file and plotting the nodes according to their degrees.

The graph database also provides information about data sharing in a more concise way, projecting results (degree, indegree, outdegree) onto tables in order to observe the data circulation in an ecosystem. Let $\mathcal{R}$ be the set of relationships that link apps with images, i.e., $\mathcal{R}$ = {RECEIVED, SENT, DOWNLOADED, CAPTURED, UNKNOWN, DELETED, UPLOADED}. Let $\mathcal{I}$ be the set of incoming relationships and $\mathcal{O}$ be the set of outgoing relationships, i.e., $\mathcal{I}$ = {RECEIVED, DOWNLOADED, CAPTURED} and $\mathcal{O}$ = {SENT, UNKNOWN, DELETED, UPLOADED}. Additionally, let $\mathcal{A}$ be the set of app sources in an ecosystem, i.e., $\mathcal{A}$ = {Bluetooth, Beam, Email, Downloads, Messenger, Drive, Gmail, Hangouts, Instagram, Dropbox, Snapchat, Camera, Facebook, Twitter, WhatsApp}. In general, $\mathcal{A} = \{A_i, i = 1, \ldots, n\}$, where n is the number of different apps in the ecosystem. If m is the number of different relationships that link app A_i, then the relationships

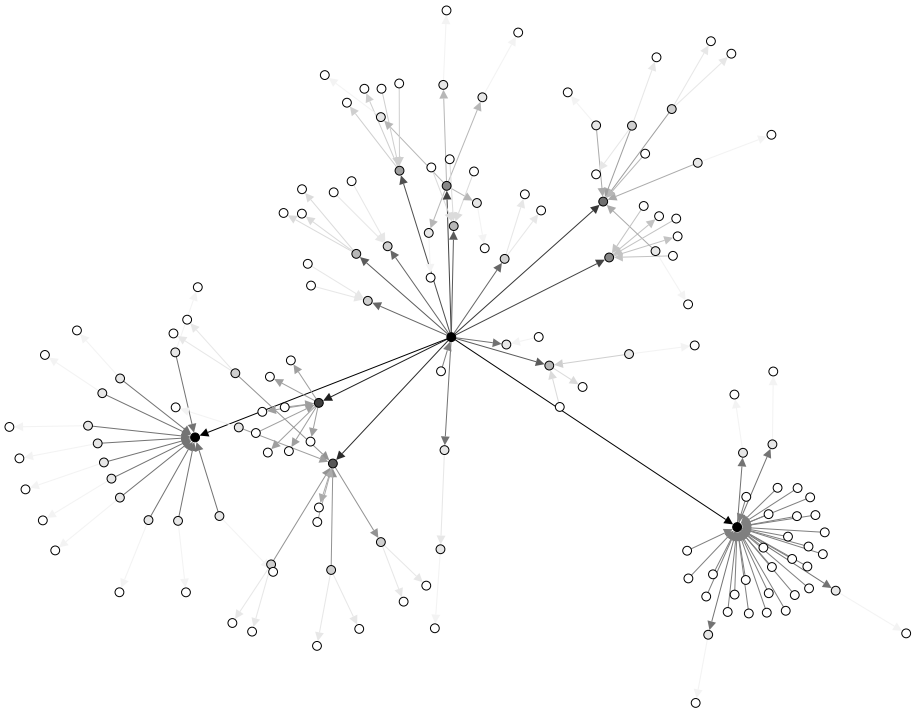


Figure 5. Graph with "busy" nodes highlighted.

may be expressed as ${r^{(i)}}_j$, $j = 1, \ldots, m$ and ${r^{(i)}}_j \in \mathcal{A}$. For simplicity, ${r^{(i)}}_j$ is expressed as:

$$ {r^{(i)}}_j = \begin{cases} \kappa & \text{if } {r^{(i)}}_j \in \mathcal{I} \\ \lambda & \text{if } {r^{(i)}}_j \in \mathcal{O} \end{cases} \tag{1} $$

Thus,

$$ indegree = \sum_{i=1} \kappa \tag{2} $$

$$ outdegree = \sum_{j=1} \lambda \tag{3} $$

$$ degree = indegree + outdegree \tag{4} $$

Table 3 shows the results of applying Equation (4) on the graph database. The results indicate that Messenger stores more information compared with the other apps. Also, the user of the smartphone exchanged images using Bluetooth and NFC (Beam) (indegree $\neq 0$). Other information can be gleaned about the use of chatting apps for image exchange

Table 3. Numbers of relationships between images and apps.

App	Indegree	Outdegree	Degree
Bluetooth	5	2	7
Beam	1	0	1
Email	2	0	2
Downloads	6	0	6
Messenger	28	4	32
Drive	2	1	3
Gmail	0	3	3
Hangouts	2	2	4
Instagram	0	5	5
Dropbox	2	6	8
Snapchat	1	0	1
Camera	11	0	11
Facebook	3	0	3
Twitter	0	2	2
WhatsApp	3	2	5

(WhatsApp and Hangouts). For example, the person under investigation cannot claim that no image left the device via WhatsApp because the outdegree for WhatsApp equals two. This means that at least two images were sent via WhatsApp to another ecosystem. Additionally, if the image nodes were to be connected to time nodes, it would be possible to see when these transactions occurred.

7. Conclusions

The framework presented in this chapter provides a novel approach for capturing and analyzing the flow of images in smartphone ecosystems. The relevant information is collected from SQLite databases in the caches and data partitions of Android smartphones (and tablets) and stored in a Neo4j graph database for forensic analyses. The graph database supports rapid and accurate searches based on pattern matching. The framework is extensible and can be adapted to provide big data functionality by adding diverse, semi-structured data from a variety of sources. An unexpected discovery during the study was that applications such as Facebook can access the smartphone Gallery app and copies of images in the particular folders are retained by the applications. This means that even if a user has deleted the original images from Gallery, the cached files corresponding to the app can reveal the deleted content.

Future research will leverage the advantages of graph databases to accommodate multiple entities in a single graph. These entities, which

could be fake or secondary social media accounts or friends, colleagues or family members in the contact list of a seized smartphone, will be linked with actions and events to produce additional evidence. For example, if an individual under investigation has exchanged via Bluetooth an image with embedded geolocation data that was taken at a specific time, then the recipient's location could also be derived from the information that is collected and organized in a graph database. Thus, the framework presented in this chapter can be extended to hold diverse data that produces additional information related to entities who interact with an individual under investigation.

Acknowledgement

This research was supported by the EU DG Home Affairs – ISEC (Prevention of and Fight Against Crime)/INT (Illegal Use of Internet) Programme (HOME/2012/ISEC/AG/INT/4000003892) and by the Systems Centre of the University of Bristol.

References

[1] P. Andriotis, G. Oikonomou and T. Tryfonas, Forensic analysis of wireless networking evidence of Android smartphones, *Proceedings of the IEEE International Workshop on Information Forensics and Security*, pp. 109–114, 2012.

[2] C. Anglano, Forensic analysis of WhatsApp Messenger on Android smartphones, *Digital Investigation*, vol. 11(3), pp. 201–213, 2014.

[3] H. Chung, J. Park, S. Lee and C. Kang, Digital forensic investigation of cloud storage services, *Digital Investigation*, vol. 9(2), pp. 81–95, 2012.

[4] A. Flaglien, A. Mallasvik, M. Mustorp and A. Arnes, Storage and exchange formats for digital evidence, *Digital Investigation*, vol. 8(2), pp. 122–128, 2011.

[5] J. Fridrich, Digital image forensics, *IEEE Signal Processing*, vol. 26(2), pp. 26–37, 2009.

[6] S. Garfinkel, Digital forensics XML and the DFXML toolset, *Digital Investigation*, vol. 8(3-4), pp. 161–174, 2012.

[7] J. Grover, Android forensics: Automated data collection and reporting from a mobile device, *Digital Investigation*, vol. 10(S), pp. S12–S20, 2013.

[8] A. Hoog, *Android Forensics: Investigation, Analysis and Mobile Security for Google Android*, Syngress, Waltham, Massachusetts, 2011.

[9] T. Howard, Don't cache out your case: Prosecuting child pornography possession laws based on images located in temporary Internet files, *Berkeley Technical Law Journal*, vol. 19, pp. 1227–1273, 2004.

[10] M. Huber, M. Mulazzani, M. Leithner, S. Schrittwieser, G. Wondracek and E. Weippl, Social snapshots: Digital forensics for online social networks, *Proceedings of the Twenty-Seventh Annual Computer Security Applications Conference*, pp. 113–122, 2011.

[11] R. Hurley, S. Prusty, H. Soroush, R. Walls, J. Albrecht, E. Cecchet, B. Levine, M. Liberatore, B. Lynn and J. Wolak, Measurement and analysis of child pornography trafficking on P2P networks, *Proceedings of the Twenty-Second International Conference on World Wide Web*, pp. 631–642, 2013.

[12] G. Kontaxis, I. Polakis, S. Ioannidis and E. Markatos, Detecting social network profile cloning, *Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 295–300, 2011.

[13] M. Liberatore, R. Erdely, T. Kerle, B. Levine and C. Shields, Forensic investigation of peer-to-peer file sharing networks, *Digital Investigation*, vol. 7(S), pp. S95–S103, 2010.

[14] Q. Liu, X. Li, L. Chen, H. Cho, P. Cooper, Z. Chen, M. Qiao and A. Sung, Identification of smartphone-image source and manipulation, in *Advanced Research in Applied Artificial Intelligence*, H. Jiang, W. Ding, M. Ali and X. Wu (Eds), Springer, Berlin-Heidelberg, Germany, pp. 262–271, 2012.

[15] F. Marturana and S. Tacconi, A machine-learning-based triage methodology for automated categorization of digital media, *Digital Investigation*, vol. 10(2), pp. 193–204, 2013.

[16] A. Mylonas, V. Meletiadis, L. Mitrou and D. Gritzalis, Smartphone sensor data as digital evidence, *Computers and Security*, vol. 38, pp. 51–75, 2013.

[17] M. Stamm and K. Liu, Anti-forensics of digital image compression, *IEEE Transactions on Information Forensics and Security*, vol. 6(3), pp. 1050–1065, 2011.

[18] S. Teelink and R. Erbacher, Improving the computer forensic analysis process through visualization, *Communications of the ACM*, vol. 49(2), pp. 71–75, 2006.

[19] P. Turner, Unification of digital evidence from disparate sources (digital evidence bags), *Digital Investigation*, vol. 2(3), pp. 223–228, 2005.

[20] J. Wolak, M. Liberatore and B. Levine, Measuring a year of child pornography trafficking by U.S. computers on a peer-to-peer network, *Child Abuse and Neglect*, vol. 38(2), pp. 347–356, 2014.