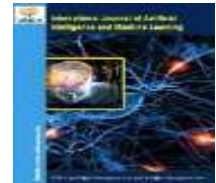




SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Self-Destructive Messages on the Android Platform: A Secondary Data-Based Study of Forensic Recovery and Cybersecurity Applications

Prince Kumar¹, Dr. Ekbal Rashid², Dr. Ritushree Narayan³

¹Research Scholar, Faculty of Computing & IT, Usha Martin University Ranchi, India.

E-mail: kprince87@gmail.com

²Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Bawrampet, Hyderabad, 500043, Telangana, India. E-mail: dr.ekbalrashid@klh.edu.in

³Assistant Professor, Faculty of Computing & IT, Usha Martin University Ranchi, India. E-mail: ritushree@umu.ac.in

Abstract

Self-destructive messaging removes visible content after a set time or viewing event, but application-level expiry does not necessarily result in storage-level erasure. This work proposes a secondary evidence-based reproducible framework for comparing the residual artefacts of self-destructive messaging on Android. The evidence base was updated through August 2026 and presented as a residual-artefact taxonomy (primary database, journal/WAL, media cache, notification, log/telemetry, backup, and volatile memory), which was then applied to six of the most frequently studied applications: Signal, Telegram, WhatsApp, Snapchat, Wickr Me, and Confide. A total of 42 application-artefact cells were coded from the literature reviewed. Based on this matrix a Residual-Artefact Persistence Index (RAPI) is calculated. RAPI is a literature-based evidence-synthesis score that categorically quantifies the frequency of reporting residual artefacts in the reviewed studies and is NOT a probability that a given device will have successful forensic recovery. To interpret the rank in relation to the coding assumptions, baseline RAPI values and a leave-one-category-out sensitivity analysis are reported as well as two alternative weighting schemes. A lower score may therefore reflect a less extensive published evidence base rather than necessarily indicating more robust secure deletion – this is highlighted for Confide. The paper is a structured secondary synthesis and controlled replication on the current Android devices is still necessary, as no new experiments were carried out with the device. The contribution is a common artefact taxonomy, a 42-observation coding framework, a comparative analysis, as well as a set of clearly separated engineering recommendations for developers and investigators.

Keywords: Android forensics, digital evidence, ephemeral messaging, mobile device forensics.

1. Introduction

Disappearing messages, view-once media, secret chats, burn-on-read, etc. are becoming increasingly common features in mobile messaging applications. These features deplete content from the common UI following a viewing event or after a timer expires; but for the forensic examiner the question is, does any related content exist in any storage layer after the application claims it's expired. Forensic practice in mobile devices thus is different from application-level deletion to secure erasure and is required to be documented in a reproducible manner in the process of acquisition, preservation, interpretation, and reporting [1]–[3].

Android doesn't make this so easy, as each communication event can generate multiple instances of the same information. A message can be written to an application database, written to a write-ahead log, rendered into a notification, stored as a thumbnail, written to a backup or kept temporarily in memory, or written to telemetry. Previous studies of mobile social apps and the acquisition of Android devices have confirmed that traces of user-interface deletion may be found in allocated, unallocated, and application-specific storage [4]–[6]. Therefore, the lack of a message in the visible conversation, does not imply the lack of any of the artefacts.

Forensic literature is technically profuse, but difficult to compare. Different versions of Android, hardware, app versions, rooted vs. unrooted phones, acquisition techniques, and commercial tools are used in studies. Some report on the content of messages, others on metadata only, and still others on encrypted databases, media caches, memory or network traces [7]–[12]. A single recovery percentage, taken from such studies and aggregated across them, would be methodologically weak, therefore, unless the denominators and conditions used in the experiments were similar.

This updated study aims to overcome this issue by using a common taxonomy of residual-artefacts and an explicit coded secondary dataset. Four contributions are new as compared to previous application specific forensic studies. First, the seven category taxonomy assigns a lifecycle of a self-destructive message to the residual storage vectors (independent from any vendor) that are valid for applications. Second, a 42 observation coding framework (six applications x seven categories) translates qualitative findings into an auditable matrix. Third, a comparative Residual-Artefact Persistence Index (RAPI) is also calculated along with a leave-one-category-out analysis and alternative weighting schemes and thus the ranking is not reported as a single number but rather as a sensitivity. Fourth, engineering recommendations for developers and investigators are isolated from the literature-supported evidence distinguishing reviewed findings from authors' inferences. RAPI is not a probability of a forensic recovery anywhere in the document but rather an evidence-synthesis score based on the literature that is reviewed

2. Related Work and Research Gap

A. Android Messaging and Application Forensics

The general Android acquisition moved to the application level forensic analysis, which reconstructed the app specific databases, preferences, media and event histories. Local databases and metadata had been shown to be useful in WhatsApp research [7] and signaling and communication context was shown to be useful using network analysis [8]. Comparative Android studies then were able to recover messages, contacts, media and other artefacts from the various social-messaging applications [9]. Taxonomy-based work also demonstrated the possibility of comparing communication applications using common classes of evidentiary artefacts in addition to app specific names of the file [12].

The other component of encrypted messaging is the fact that an artefact may still exist physically but be unreadable until the key material and/or application logic is available. In the case of ChatSecure, the research revealed that encrypted local databases and volatile memory key material can be correlated to recreate communication [13]. The studies of Telegram identified application databases, media structures and chronology of messages [14], [17]. Researches of WeChat and Android vault applications also revealed that even after deletion or protection, data can still be reconstructed based on the level of data acquisition and the state of the data stored [18], [19].

B. Ephemeral and Privacy-Oriented Messaging

Fewer directly comparable experiments have been tried with ephemeral messaging. Azhar and Barton focused on the analysis of Wickr and Telegram on Android for ephemeral and secure operations with a consideration of artefacts in the database and memory [15]. Although snaps are supposed to disappear, Alyahya and Kausar demonstrated that forensic artefacts are still left on Android devices when using Snapchat [16]. A larger study of ephemeral apps on Android and iOS identified app- and account-specific artefacts, media stored in the cache, and significant differences between the various applications; the evidence on Android was significantly less than several other apps, especially Confide [24].

New research clarifies the difference between encryption and deletion. Later work, which demonstrated the database and multimedia decryption of Wickr [26], decrypted database, multimedia, log, and preference files for both Signal and Wickr on Android [28]. These studies provide proof of recoverability of protected local data but are not in and of themselves a post-expiry test. Heath, MacDermott and Akinbi directly assessed the disappearing message on WhatsApp, Snapchat and Telegram on mobile, desktop and cloud platforms and found there was a high reliance on platform and extraction method [29]. Later, Sudiana et al. managed to retrieve 83.33% of the evidence of WhatsApp disappearing messages in six specific cases of unrooted-Android devices from backup files and notification logs [30].

C. Research Gap

Three gaps remain. Firstly, artefacts reported in application studies are reported in different terms, which makes it difficult to compare between different applications. Secondly, the literature tends to equate message recovery with metadata recovery and simply having application artefacts. Third, the new studies highlight the persistence of evidence following privacy-focused activities: In 2026, a systematization of knowledge showed 93 residual observations from 18 mobile anti-forensics studies, mostly focusing on application-private and operating-system-level storage [31]; a study on the WhatsApp view-once feature in 2026 correlated database metadata and a remaining thumbnail with the source media following physical extraction [32]. Such results suggest an evidence-based rather than absolute sense of persistence in a category-based synthesis.

3. Android Storage and Residual-Artefact Model

A. Message Lifecycle and Storage Vectors

There is a self-destructive message that goes through composition, transmission, receipt, rendering, expiry, deletion and potential residual existence. Artefacts can be produced outside the controlled site, at each stage. Message rows, metadata in application database, older pages in a SQLite journal; a media pipeline may create previews, content in notification on operating system, identifiers in crash or diagnostic logs, backup copies can survive the visible content, recently decrypted content may exist in volatile memory. The availability of these vectors depends on the Android version, version of the application, encryption state, device configuration, and acquisition time [1], [6], [11], [22]. Table 1 lists the analytical dimensions that were used to organize the secondary data for this residual-artefact model.

Table 1. Analytical Dimensions Used In the Secondary Dataset

Dimension	Representative artefact	Typical locus	Acquisition emphasis
Primary DB	Rows, freelist or metadata	App-private DB	Logical / physical
Journal/WAL	Old pages, rollback/WAL	Beside primary DB	WAL replay / carving
Media cache	Thumbnail or preview	Cache / media store	Logical / physical
Notification	Displayed text / metadata	OS notification state	Privileged / device
Log/telemetry	Logs, crash/diagnostic data	App / vendor log path	Logical / privileged
Backup	Local or cloud copy	Host / backup service	Host / cloud
Memory	Recently decrypted content	RAM / swap-like state	Live acquisition

B. Persistence Semantics

It is worth noting that in the taxonomy there is a clear distinction between 'retained' and 'partially retained'. Retained: Reviewed evidence consistently indicates a residual that can be recovered in that category under documented conditions. Partial: the evidence is metadata, tool-specific, version specific, conditional, or otherwise very closely related to but not a direct post-expiry experiment. A dash indicates that the category was not mentioned in any of the sources examined, and should not be interpreted to mean that the artefact has no existence. This distinction is needed because the claim of secure deletion cannot be made from the non-observation in a diverse secondary literature.

4. Materials and methods

A. Secondary-Evidence Selection

The updated evidence base includes a targeted update search that was conducted on 18 August 2026 in addition to the original manuscript bibliography. The databases and indexing services searched were IEEE Explore, Elsevier Science Direct, ACM Digital Library, Springer Link, Wiley Online Library and Google Scholar. Search strings combined the term Android AND (forensic OR forensics) AND ('ephemeral messaging' OR 'disappearing messages' OR 'self-destructive messages' OR 'view-once' OR 'secret chat' OR 'expiring messages') and were applied to titles, abstracts, and author keywords. The time period for the search was set to 2012-2026 to cover the baseline Android forensic research and post-expiry research. A total of 214 different records were retrieved after de-duplication (IEEE 46, Science Direct 58, ACM 27, Springer Link 31, Wiley 12, Google Scholar snowball 40). At the title and abstract screening stage, non-Android platforms, non-messaging application, editorials and short abstracts, were excluded and 68 records were advanced for full-text assessment. For full-text screening, the inclusion criteria were a target of Android and a mention of the application or the underlying layer of storage which was informative in relation to the six-target applications, in addition to the mention of artefacts at one or more of the seven remaining categories. Studies that lacked

methodological description, those without any results of acquisition that were only of a theoretical or legal nature, and the studies that readdressed earlier ones included were excluded. Thirty two records met all criteria and are the coded evidence base that is cited in the reference list. A backward citation search was carried out on 32 of the records but did not identify additional studies that would meet the inclusion criteria. Due to the diversity in source studies in terms of device, tool and expiry mechanisms, the analysis is a structured secondary evidence synthesis – it was not possible to conduct a statistical meta-analysis for which a pooled recovery percentage is calculated.

The unit of analysis is the category of the application-artefact. There are 42 coded observations in the six applications and seven categories which enable tracing of each quantitative figure in the results section to an entry in the coding matrix (Table 3) and to the supporting reference(s) recorded in the evidence-provenance table (Table 3-C).

B. Evidence Coding and Provenance

The core empirical evidence that was used to audit each application profile is noted in Table 2. The table also differentiates the evidence that is based directly on the expiry from that of studies where the main aim was to enable recovery of locally saved or encrypted artefacts. This stops a decryption result, for instance, being displayed as an automatic disappearance-message result.

Table 2. Core Evidence Basis by Application

Application	Core sources	Evidence emphasis
Signal	[24], [28]	Encrypted DB/media/log artefacts; expiry-specific evidence remains limited
Telegram	[14], [15], [17], [29]	DB, media and ephemeral/secret-chat artefacts; tool-dependent post-expiry findings
WhatsApp	[7], [21], [27], [29], [30], [32]	DB/metadata, backup, notification and view-once traces
Snapchat	[16], [24], [29]	Ephemeral media, cache and platform-dependent recovery
Wickr Me	[15], [26], [28]	Ephemeral storage plus encrypted DB/media/log recovery
Confide	[24]	Account/configuration artefacts; limited conversation-level Android evidence

C. Residual-Artefact Persistence Index

For each application a and artefact category c , the coding score $s(a,c)$ is 1.0 for retained (R), 0.5 for partial/conditional (P), and 0 for not seen in the literature reviewed for each application a and artefact category c . The baseline index gives all seven categories the same weight. Thus, RAPI is a normalized literature based evidence score. It is not a probability that a specific deleted message will be recovered from a live device and should not be used as a recovery rate.

$$RAPI(a) = (1 / 7) \times \sum s(a,c), \text{ for } c = 1, \dots, 7$$

In this manuscript, the numbers for the arithmetic have been recomputed from the coded matrix, rather than manually transcribed. Leave one category out (LOO) analysis is conducted seven times for each application with one category being left out on each iteration and minimum and maximum values are reported. Two other weightings are also given (6-C). In scheme W1, the weight of primary database and journal/WAL categories is doubled as these are the categories that directly contain the message content. In scheme W2, the weight of notification, media cache, and backup categories is doubled. This is as these categories are most commonly accessible without root or specialist acquisition. Both schemes are normalised, that is per-application indices are in [0,1] and can be compared with the baseline.

5. Comparative Application Analysis

A. Signal and Telegram

Signal protects locally stored information using strong encryption and Android key-management mechanisms. Recent forensic research, however, indicates that certain acquisition circumstances will allow encrypted database, multimedia, log, and preference files to be decrypted [28]. For self-destructive messaging, the implication is not that all out-of-date Signal messages can be recovered, but that acquisition shouldn't be limited to just the conversation view, and ancillary artefacts need to be studied. However, Telegram offers a larger persistence surface as both the local databases and the

media structures are well documented [14], [17] and experiments with secure or ephemeral functions have resulted in recoverable artefacts under specific conditions [15], [29].

B. WhatsApp and Snapchat

WhatsApp offers disappearing-message timers, as well as view-once media. The evidence base includes application databases, network metadata, backups, notification logs and recent view-once correlation [7], [8], [27], [30], [32]. Notably, results are acquisition-dependent: Heath et al. observed significant differences between combinations of devices and tools [29], while Sudiana et al. identified artefacts of the disappearing message in five out of six scenarios through backup and notification artefacts [30]. Although Snapchat's design is essentially event-driven, there are recoverable artefacts and traces of media related activities in Android [16], [24]. Heath et al. also demonstrated that they were able to obtain significant evidence on the Android under their conditions on Snapchat [29].

C. Wickr Me and Confide

Although Wickr Me is supposed to be used to send messages that cannot be recovered, a range of local artefacts such as multimedia and databases have been recovered or decrypted when appropriate methods for acquisition and analysis have been available [15], [26], [28]. The evidence published by Confide is thin in the Android world. In a cross-platform investigation conducted in 2020, the artefacts associated with the configuration and accounts were recovered on Android but no complete conversation reconstruction was reported after the conversation is over [24]. Confide's relatively low RAPI value should thus be interpreted largely as a consequence of its limited published evidence, rather than as proof of more robust secure deletion.

D. Coded Comparative Dataset

The entire 42-cell set of data is displayed in two convenient panels in Table 3. The matrix was compiled from persistence trends reported and represents a comparative triage. It does not cancel out study-specific conditions, and should be read in conjunction with Table 2 and the source-specific benchmarks in Section VI. The coded persistence matrix is summarized visually by colour in Fig. 1.

Table 3-A. Coded Persistence Matrix: Signal, Telegram and Whatsapp (R=1, P=0.5, --=0).

Supporting References Per Cell Are Given In Table 3-C.

Artefact	Signal	Telegram	WhatsApp
Primary DB	P	R	R
Journal/WAL	P	R	R
Media cache	P	R	R
Notification	R	R	R
Log/telemetry	P	P	P
Backup	R	R	R
Memory	R	R	R

Table 3-B. Coded Persistence Matrix: Snapchat, Wickr Me and Confide (R=1, P=0.5, --=0).

Supporting References Per Cell Are Given In Table 3-C.

Artefact	Snapchat	Wickr Me	Confide
Primary DB	P	—	—
Journal/WAL	P	—	—
Media cache	R	P	P
Notification	R	R	R
Log/telemetry	P	R	R
Backup	P	—	—
Memory	R	R	P

Table 3-C. Evidence Provenance for the Coded Persistence Matrix (Supporting References Per Cell). A Dash (--) Indicates The Reviewed Literature Did Not Report An Artefact In That Category.

Applicati on	Prim ary DB	Journal /WAL	Medi a cach e	Notifi catio n	Log/tel emetry	Backu p	Memor y
Signal	[28]	[28]	[28]	[29]	[28]	[28]	[28]
Telegram	[14], [17]	[17]	[17]	[17], [29]	[15], [17]	[17]	[15], [17]
WhatsApp	[7], [27], [30]	[7],[22]	[7], [32]	[29], [30]	[8]	[7],[30]	[21]
Snapchat	[16]	[16]	[16], [24]	[29]	[16]	[24]	[24]
Wickr Me	--	--	[15], [26]	[15]	[28]	--	[15], [26]
Confide	--	--	[24]	[24]	[24]	--	[24]

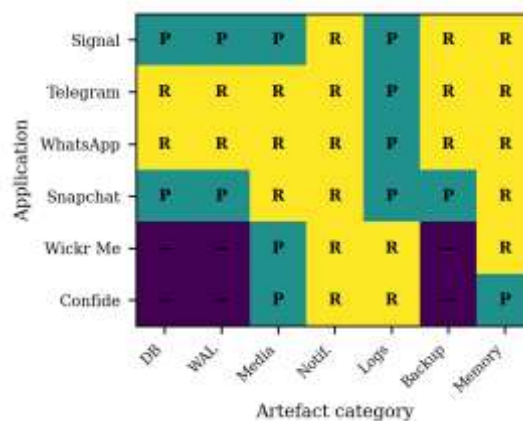


Fig. 1. Evidence-coded residual-artefact persistence across six applications and seven categories, based on the reviewed literature. Cells shaded for R (retained), P (partial/conditional), and -- (not observed). Source: authors' coding of the secondary dataset.

6. Result and evaluation

A. Corrected Application-Level Index

There are 23 observations of R, 13 observations of P, and six categories that are not observed in the reviewed evidence in the coded matrix. The average of the normalized scores in all 42 cells is 0.702. Telegram and WhatsApp have the highest RAPI values (0.929 each) in the coded dataset. Signal and Snapchat are tied at 0.714; Wickr Me is 0.500 and Confide is 0.429. These values are literature-derived evidence-synthesis values, which are an overview of the prevalence of residual artefacts in each of the seven categories for every application. They do not correspond to the chances of being able to reconstruct a particular expired message from a particular device. The baseline RAPI as well as the minimum and maximum LOO is reported in Table 4. The baseline RAPI values corresponding to the same are plotted graphically in Fig. 2.

Table 4. Residual-Artefact Persistence Index (Rapi): Baseline, Leave-One-Category-Out (Loo) Min/Max, And Alternative Weighting Schemes W1 and W2.

Application	RAPI (baseline)	LOO min	LOO max	RAPI (W1)	RAPI (W2)
Telegram	0.929	0.917	1	0.944	0.95
WhatsApp	0.929	0.917	1	0.944	0.95
Signal	0.714	0.667	0.75	0.667	0.75
Snapchat	0.714	0.667	0.75	0.667	0.75
Wickr Me	0.5	0.417	0.583	0.389	0.5
Confide	0.429	0.333	0.5	0.333	0.45

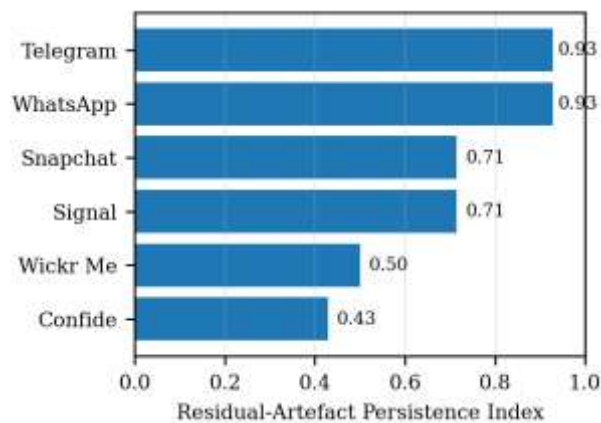


Fig. 2. Baseline Residual-Artefact Persistence Index (RAPI) by application. RAPI is a literature-based evidence-synthesis score in [0,1], not a probability of forensic recovery. Source: authors' coding of the secondary dataset.

B. Artefact-Category Pattern

The highest normalized score in the coded synthesis (1.000) was for notification records, coded in 6/6 reviewed application profiles. All the profiles reviewed had a code for volatile memory (retained or partial) with a value of 0.917. At 0.750 is the media cache. Score for log/telemetry is 0.667 while the score for primary database and journal/WAL is 0.500, and for backups is 0.583. These values represent the concentration of documented traces of evidence in the literature in terms of the six application profiles – they are not and should not be interpreted as recovery rates for any specific Android device. The normalized persistence pattern at the category-level is shown in Fig. 3.

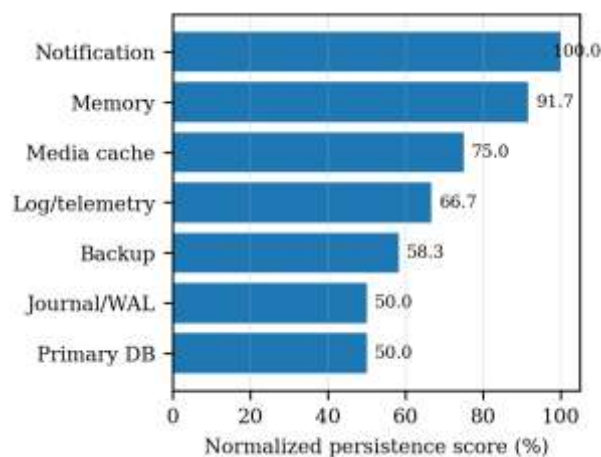


Fig. 3. Normalized persistence score by artefact category across the six application profiles. Values reflect frequency of reported residual evidence in the reviewed literature. Source: authors' coding of the secondary dataset.

C. Robustness to Category Omission

Leave-One-Category-Out (LOO) analysis is a method that checks the equal weight assumption without using arbitrary other weights. In all seven omissions, Telegram and WhatsApp are still in the top tier with recalculated scores between 0.917 and 1.000. Signal and Snapchat are tied in the middle tier, in the 0.667–0.750 range. Wickr Me varies from 0.417 to 0.583, and Confide from 0.333 to 0.500. Using weighting scheme W1 (double weight for primary database, journal/WAL and sum-normalized), the results of Telegram and WhatsApp are 0.944 each, Signal 0.667, Snapchat 0.667, Wickr Me 0.389 and Confide 0.333; penalizing ephemeral applications whose evidence basis for the two content-bearing categories are coded P or --. The results of scheme W2 (double weight on notification, media cache, and backup, sum-normalized) are: 0.950 for Telegram and WhatsApp, 0.750 for Signal, 0.750 for Snapchat, 0.500 for Wickr Me, and 0.450 for Confide. The ordering Telegram = WhatsApp > Signal, Snapchat > Wickr Me > Confide is maintained in all three schemes, demonstrating that it is fairly resilient to sensible deviations from the weighting assumption. The sensitivity range for each of the baselines RAPIs, where one category is omitted is shown in Fig. 4.

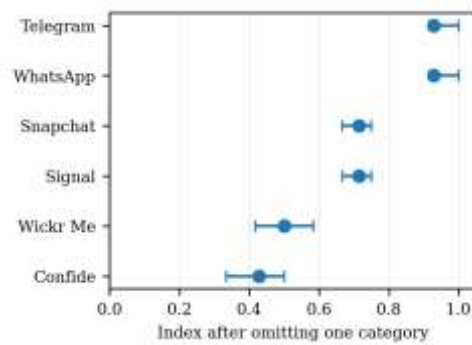


Fig. 4. Leave-one-category-out sensitivity range around each baseline RAPI. Whiskers show the minimum and maximum RAPI when one artefact category at a time is removed from the seven-category index. Source: authors' coding of the secondary dataset.

D. Cross-Check against Recent Empirical Studies

Table 5 intentionally shows the latest empirical results at the level they were measured. However, the results of these are not consolidated into the index since they are generated by different devices, different OS, different extractors, different expiry policies, and different metrics for successful extraction. This treatment is source specific as opposed to assigning a generic recovery percentage to an application, and is more defensible.

Table 5. Recent Empirical Cross-Checks and Their Proper Interpretation

Study	Platform / scope	Key finding and interpretation
Heath et al. [29]	WhatsApp, Snapchat, Telegram; mobile/desktop/cloud	Post-expiry results differed materially by app, device and extraction path; treated as condition-dependent evidence.
Sudiana et al. [30]	WhatsApp; unrooted Android; six scenarios	83.33% of tested evidence was recovered via backup/notification traces; used only as a scenario-specific benchmark.
Son et al. [28]	Signal and Wickr; rooted/unrooted Android	Encrypted DB, multimedia, log and preference files were decrypted; supports residual storage, not a direct expiry rate.
Okoye et al. [31]	197-study systematization; 18 mobile studies	Reported 93 mobile residual observations; independently supports multi-layer persistence.
Raja [32]	WhatsApp view-once; Android physical extraction	Message metadata and a surviving thumbnail supported media correlation after view-once transmission.

7. Cybersecurity Application and Implications

A. Forensic Triage and Incident Response

Comparative results suggest that a storage-vector triage approach is warranted. Investigators need to preserve volatile data if it is lawful and technically feasible to do so, acquire application-private storage before any app activity is likely to change the data, and analyze journals, caches, notification records, and backups instead of using the primary database as the only source of evidence. If a commercial parser fails to produce any messages, this does not mean that you have no data; it may instead mean that the parser has failed, so tool output should be validated against raw data or against artifacts which can be independently interpreted [1], [22], [23].

The same framework can enhance incident response for organizations in the event of insider data leakage, harassment, and fraud and policy violations. Even when content expires from the application, endpoint/mobile-device management telemetry can identify notification/log metadata. But it shall be proportionate and lawfully authorized and recorded, to preserve its evidentiary value without falling into the trap of uncontrolled surveillance.

B. Privacy Engineering and Secure Deletion

The results for the developers detect specific areas of the residual-data surface as reduction points. Where possible, expiry should cause coordinated clean-up for database state, journal, cached media, notification content, local/cloud backups, and telemetry and sensitive in-memory buffers. The aim is not to simply delete a row from the user interface, but to reduce any superfluous representations and to shorten their retention times. These developer mitigation priorities outlined by the artefact model is summarized in Table 6.

Table 6. Developer Mitigation Priorities Suggested By the Artefact Model

Mitigation	Targeted residue	Expected effect	Priority
Checkpoint / compact database safely	Journal/WAL and stale rows	Reduces older page versions after expiry	High
Content-minimized notifications	Notification history	Prevents duplicate plaintext exposure	High
Expiry-aware backup policy	Local/cloud backup	Limits retention beyond user-facing expiry	High
Cache lifecycle enforcement	Media previews/thumbnails	Reduces recoverable derivative media	High
Sensitive log redaction	Log/telemetry	Removes identifiers/content from diagnostics	Medium
Memory hygiene after use	Volatile memory	Shortens lifetime of decrypted content	Medium

C. Interpretation and Legal Defensibility

A residual artefact should be specified, where it came from, how it was acquired, what its timestamp semantics are, and what are its known limitations. In cases where the content is reconstructed from a journal, thumbnail, notification or backup and not directly from the live application database, the provenance is relevant to the reproduction of the evidence, the chain of custody and technical interpretation of the evidence. While suitable for use as a prioritization tool for triage and reporting, the use of RAPI does not replace jurisdiction-specific determination of admissibility which is based on rules of evidence, disclosure and expert-witness practice beyond the scope of this study.

8. Threats to Validity

A. Internal and Construct Validity

The main constraint is reliance on secondary evidence published elsewhere. There are differences between source studies related to the model of the handset, Android release, version of the application, type of message, and time from expiry to acquisition, root status, and tool support. The R/P/--classification is a ordinal classification which groups those conditions together. Also the equal weighting assumes that the contribution of a notification trace and a primary database record to the persistence of the residual-artefact are equal, which is simplified; the other weighting schemes in Section VI-C partially address this, but not completely. It is important to remember that RAPI is a summary of the literature reviewed and not an inherent property of any application: the ranking is a summary of what has been reported and a claim that one application is more or less recoverable than another is not being made.

B. External and Conclusion Validity

Application behavior is very dynamic. Older file paths or recovery procedures can become invalid due to scoped storage, encryption changes, database migrations, cloud-synchronization policies and parser updates. In comparison to other messaging apps like WhatsApp, Telegram and Snapchat, there is very little published evidence for Confide. This means that Confide's lower RAPI may be due to a thinner published evidence base and/or more effective secure deletion. This is true for any class where a controlled post-expiry test hasn't been published in the literature we've read. Therefore, results presented must be considered to be up-to-date until the latest literature is reviewed, in August 2026, and re-coding would occur based on future controlled experiments.

9. Conclusion and future work

Self-destructive messaging is not a binary forensic state. Expiry means that the content is removed from the ordinary interface but related traces may be kept in databases, journals, caches, notification records, logs, backups and/or volatile memory. This study transforms the literature reviewed into a 42-observation coded dataset, thus allowing the claims to be audited and providing reproducible figures and ranks. Throughout, RAPI is reported as a literature-based evidence-synthesis score, and not as a probability of successful forensic recovery. The ranking is consistent across both leave-one-category-out and two different schemes of weights; however a lower score does not necessarily indicate a better secure-deletion property, but rather a more sparse published evidence base, which is highlighted for Confide.

Future work should be carried out using controlled, versioned experiments involving more recent versions of Android and the same workloads on messages. The same devices should be acquired at regular times, following their expiry, through logical, file system, physical (where lawful), backup, and volatile-memory methods. The reporting should cover both message-content recovery and metadata-only recovery and the coding matrix in Table 3 should be re-evaluated as per the primary results. The engineering recommendations provided here (Table 6) are the authors' interpretations from the literature review, and should be tested with controlled measurements before generalization.

10. References

- [1] National Institute of Standards and Technology, Guidelines on Mobile Device Forensics, NIST Special Publication 800-101 Rev. 1, 2014.
- [2] A. Hoog, *Android Forensics: Investigation, Analysis and Mobile Security for Google Android*. Syngress, 2011.
- [3] E. Casey, *Digital Evidence and Computer Crime*, 3rd ed. Academic Press, 2011.
- [4] N. Al Mutawa, I. Baggili, and A. Marrington, "Forensic analysis of social networking applications on mobile devices," *Digital Investigation*, vol. 9, pp. S24–S33, 2012.
- [5] K. Barmapsalou, D. Damopoulos, G. Kambourakis, and V. Katos, "A critical review of 7 years of mobile device forensics," *Digital Investigation*, vol. 10, no. 4, pp. 323–349, 2013.
- [6] N. Son, Y. Lee, D. Kim, J. I. James, S. Lee, and K. Lee, "A study of user data integrity during acquisition of Android devices," *Digital Investigation*, vol. 10, pp. S3–S11, 2013.
- [7] C. Anglano, "Forensic analysis of WhatsApp Messenger on Android smartphones," *Digital Investigation*, vol. 11, no. 3, pp. 201–213, 2014.
- [8] F. Karpisek, I. Baggili, and F. Breitingner, "WhatsApp network forensics: Decrypting and understanding the WhatsApp call signaling messages," *Digital Investigation*, vol. 15, pp. 110–118, 2015.
- [9] D. Walnycky, I. Baggili, A. Marrington, J. Moore, and F. Breitingner, "Network and device forensic analysis of Android social-messaging applications," *Digital Investigation*, vol. 14, pp. S77–S84, 2015.
- [10] C. Sgaras, M. T. Kechadi, and N. A. Le-Khac, "Forensics acquisition and analysis of instant messaging and VoIP applications," in *Computational Forensics*, Springer, 2015, pp. 188–199.
- [11] S. J. Yang, J. H. Choi, K. B. Kim, and T. Chang, "New acquisition method based on firmware update protocols for Android smartphones," *Digital Investigation*, vol. 14, pp. S68–S76, 2015.
- [12] A. Azfar, K.-K. R. Choo, and L. Liu, "An Android communication app forensic taxonomy," *Journal of Forensic Sciences*, vol. 61, no. 5, pp. 1337–1350, 2016, doi: 10.1111/1556-4029.13164.
- [13] C. Anglano, M. Canonico, and M. Guazzone, "Forensic analysis of the ChatSecure instant messaging application on Android smartphones," *Digital Investigation*, vol. 19, pp. 44–59, 2016, doi: 10.1016/j.diin.2016.10.001.
- [14] G. B. Satrya, P. T. Daely, and M. A. Nugroho, "Digital forensic analysis of Telegram Messenger on Android devices," in *Proc. Int. Conf. Information and Communication Technology and Systems*, 2016, doi: 10.1109/ICTS.2016.7910263.
- [15] M. A. H. Bin Azhar and T. E. A. Barton, "Forensic analysis of secure ephemeral messaging applications on Android platforms," in *Global Security, Safety and Sustainability—The Security Challenges of the Connected World*, Springer, 2017, pp. 27–41, doi: 10.1007/978-3-319-51064-4_3.
- [16] T. Alyahya and F. Kausar, "Snapchat analysis to discover digital forensic artefacts on Android smartphone," *Procedia Computer Science*, vol. 109, pp. 1035–1040, 2017, doi: 10.1016/j.procs.2017.05.421.
- [17] C. Anglano, M. Canonico, and M. Guazzone, "Forensic analysis of Telegram Messenger on Android smartphones," *Digital Investigation*, vol. 23, pp. 31–49, 2017, doi: 10.1016/j.diin.2017.09.002.

- [18] S. Wu, Y. Zhang, X. Wang, X. Xiong, and L. Du, "Forensic analysis of WeChat on Android smartphones," *Digital Investigation*, vol. 21, pp. 3–10, 2017.
- [19] X. Zhang, I. Baggili, and F. Breiting, "Breaking into the vault: Privacy, security and forensic analysis of Android vault applications," *Computers & Security*, vol. 70, pp. 516–531, 2017.
- [20] K. Barmpatsalou, T. Cruz, E. Monteiro, and P. Simoes, "Current and future trends in mobile device forensics: A survey," *ACM Computing Surveys*, vol. 51, no. 3, pp. 1–31, 2018.
- [21] K. Rath, U. Karabiyik, T. Aderibigbe, and H. Chi, "Forensic analysis of encrypted instant messaging applications on Android," in *Proc. 6th Int. Symp. Digital Forensic and Security*, 2018.
- [22] S. Nemetz, S. Schmitt, and F. Freiling, "A standardized corpus for SQLite database forensics," *Digital Investigation*, vol. 24, pp. S121–S130, 2018.
- [23] H. Pieterse, M. Olivier, and R. van Heerden, "Smartphone data evaluation model: Identifying authentic smartphone data," *Digital Investigation*, vol. 24, pp. 11–24, 2018.
- [24] M. A. H. Bin Azhar, R. Cox, and A. Chamberlain, "Forensic investigations of popular ephemeral messaging applications on Android and iOS platforms," *International Journal on Advances in Security*, vol. 13, nos. 1–2, pp. 41–53, 2020.
- [25] D. Kim and S. Lee, "Study of identifying and managing the potential evidence for effective Android forensics," *Forensic Science International: Digital Investigation*, vol. 33, art. 200897, 2020.
- [26] G. Kim, S. Kim, M. Park, Y. Park, Y. Lee, and J. Kim, "Forensic analysis of instant messaging apps: Decrypting Wickr and private text messaging data," *Forensic Science International: Digital Investigation*, vol. 37, art. 301138, 2021, doi: 10.1016/j.fsidi.2021.301138.
- [27] D. Wijnberg and N. A. Le-Khac, "Identifying interception possibilities for WhatsApp communication," *Forensic Science International: Digital Investigation*, vol. 38, art. 301132, 2021.
- [28] J. Son, Y. W. Kim, D. B. Oh, and K. Kim, "Forensic analysis of instant messengers: Decrypt Signal, Wickr, and Threema," *Forensic Science International: Digital Investigation*, vol. 40, art. 301347, 2022, doi: 10.1016/j.fsidi.2022.301347.
- [29] H. Heath, Á. MacDermott, and A. Akinbi, "Forensic analysis of ephemeral messaging applications: Disappearing messages or evidential data?" *Forensic Science International: Digital Investigation*, vol. 46, art. 301585, 2023, doi: 10.1016/j.fsidi.2023.301585.
- [30] D. Sudiana, C. H. Nuruddin, M. Rizkinia, and D. Husna, "Forensic analysis of WhatsApp disappearing message on unrooted Android using mobile device forensics methodology NIST SP 800-101r1," *Evergreen*, vol. 11, no. 1, pp. 516–524, 2024, doi: 10.5109/7172316.
- [31] U. Okoye, R. Karim, A. R. Onik, and I. Baggili, "SoK: Anti-forensics evolution, platform coverage, and mobile residual evidence," *Forensic Science International: Digital Investigation*, vol. 57, suppl., art. 302127, 2026, doi: 10.1016/j.fsidi.2026.302127.
- [32] A. A. Raja, "Forensic correlation of WhatsApp view-once video artefacts with device-resident source media using SQLite metadata analysis," *Journal of Computing & Biomedical Informatics*, vol. 11, no. 1, 2026, doi: 10.56979/1101/2026/1406.