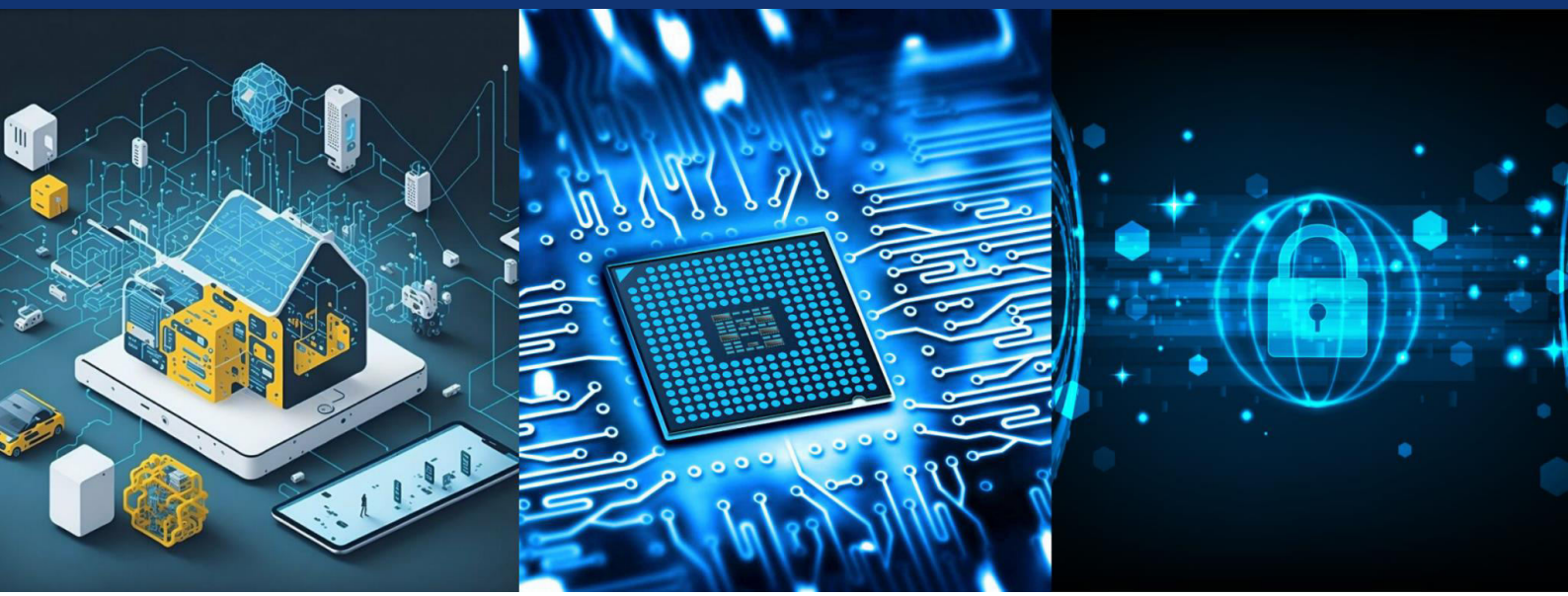
IJIRCCE

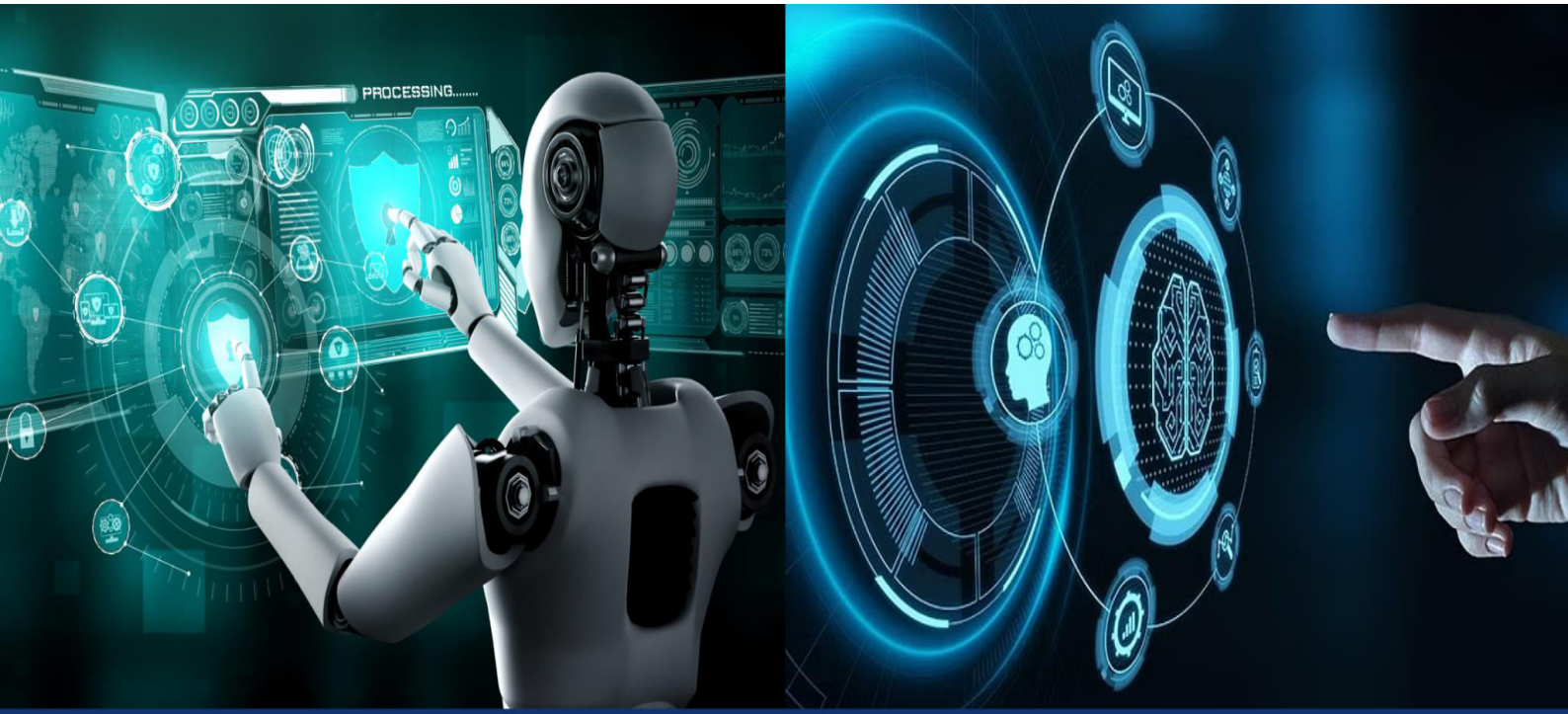
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 6, June 2026

© 2026 IJIRCCE | Volume 14, Issue 6, June 2026 DOI: 10.15680/IJIRCCE.2026.1406034

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Cloud Track: Automated Cloud Forensic Snapshot Tool for Extracting Logs, Metadata, and Deleted File Artefacts from Cloud Storage Platforms

Thowfic Ibrahim Fazil M, Rajadhurai S

PG Student, Dept. of C.F.I.S, Dr. M.G.R. Educational and Research Institute, Chennai, India

Dept. of Cyber Security, Dr. MGR Educational and Research Institute, Chennai, India

ABSTRACT: Cloud storage platforms like Google Drive are where people store a lot of evidence these days. Google Drive has over three billion users. This makes it a key place to look for evidence in investigations. The problem is that the tools we use to look at evidence like FTK, Autopsy and EnCase were made for looking at evidence on storage media. They do not work well with storage. They also do not have open-source support for getting evidence from the cloud. This is why we made Cloud Track. Cloud Track is an open-source tool that uses the Google Drive API to look at all the files on Google Drive. It checks the files to make sure they have not been changed. It does this by using something called SHA-256. It also keeps a record of what has been done to the files. This record is like a chain of custody. Cloud Track can also look for files that're not normal. It does this by using something called an Isolation Forest. This helps us look at a lot of files quickly. We can also export the evidence in a way that's safe and secure. We use something called AES-256-GCM to encrypt the evidence. We have an interface that makes it easy to use Cloud Track. The interface has eight tabs that help us investigate. We have tested Cloud Track with, up to 1,000 files. It works well. It can find the files we are looking for 94% of the time. It never finds something that's not there. Cloud Track follows all the rules for looking at evidence. It follows the NIST SP 800-86 model. It also follows the volatility-ordering requirements of RFC 3227. This means that the evidence we get from Cloud Track can be used in court. Cloud storage platforms like Google Drive are where people store a lot of evidence these days. Google Drive has over three billion users. This makes it a key place to look for evidence in investigations. The problem is that the tools we use to look at evidence like FTK, Autopsy and EnCase were made for looking at evidence on storage media. They do not work well with storage. They also do not have open-source support for getting evidence from the cloud. This is why we made Cloud Track. Cloud Track is an open-source tool that uses the Google Drive API to look at all the files on Google Drive. It checks the files to make sure they have not been changed. It does this by using something called SHA-256. It also keeps a record of what has been done to the files. This record is like a chain of custody. Cloud Track can also look for files that're not normal. It does this by using something called an Isolation Forest. This helps us look at a lot of files quickly. We can also export the evidence in a way that's safe and secure. We use something called AES-256-GCM to encrypt the evidence. We have an interface that makes it easy to use Cloud Track. The interface has eight tabs that help us investigate. We have tested Cloud Track with, up to 1,000 files. It works well. It can find the files we are looking for 94% of the time. It never finds something that's not there. Cloud Track follows all the rules for looking at evidence. It follows the NIST SP 800-86 model. It also follows the volatility-ordering requirements of RFC 3227. This means that the evidence we get from Cloud Track can be used in court.

KEYWORDS: Cloud Forensics, Digital Evidence, Google Drive, Chain of Custody, Isolation Forest, Anomaly Detection, AES-256-GCM, SHA-256, HMAC, Python, SQLite, Machine Learning

I. INTRODUCTION

Cloud platforms are where we find most of the evidence in a lot of criminal and civil investigations. Google Drive has over three billion users and it stores files that can be very important like secret documents and records of things that are against the law. We used to use tools like FTK, Autopsy and EnCase to look at files on computers. These tools are not very good at looking at files that are stored on cloud platforms like Google Drive.

These tools do not have a way to get information from cloud platforms like who can see a file and when it was changed. This makes it hard for people who investigate crimes to get the evidence they need from cloud platforms. They must use ways to get this information, or they must use special tools that are just for one cloud platform.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Cloud platforms are where the digital evidence is and we need to be able to get this evidence in a way that's reliable and works every time. Cloud platforms, like Google Drive have a lot of information that investigators need to solve crimes.

There are five problems that led to the development of Cloud Track. First there is no tool that can get all the evidence from Google Drive in a way that is good enough for a court. This means using the Google Drive application programming interface to get all the evidence. Second the ways people are doing this now like using Google Takeout exports and the GAM command-line utility do not keep a record of everything that was done to the evidence. This record is very important for a court to say that the evidence is real like what Casey and the National Institute of Standards and Technology say it should be. Third there is no tool that can automatically look at the evidence from Google Drive and find the things that might be suspicious. This means investigators must do this work by hand which can be very slow and not always right. Fourth there is no way to compare evidence from different cases to see if there are any connections. This makes it hard to find links between cases. Fifth when evidence is exported it is not protected, so someone could change it without anyone knowing. This is a problem because it means the evidence might not be real when it is used in court and this is what Cloud Track is trying to solve the problems, with Google Drive evidence. Cloud Track fixes the problems with an open-source Python tool for forensic analysis.

The system uses Google Drive API version 3 to list and check files one by one. It also uses a code called HMAC-SHA-256 to keep track of who handled the files and when. The main goal of Cloud Track is to make sure everything is done correctly and safely from the start. This means that safety is built into every part of the system. The system uses OAuth 2.0 to keep users AES-256-GCM to protect the files.

This paper will explain the following: Section II talks, about work that has been done before. Section III explains how Cloud Track works and how it was made. Section IV shows the results of testing Cloud Track. Section V. Suggests what to do next with Cloud Track.

II. RELATED WORK

The chain of custody is a legal requirement for evidence admissibility in a court of law as described by Casey [1]. Chain of Custody is a paper trail that shows where the evidence was obtained, where it was stored and who had access to it. This was codified into a four-phase forensic model — collection, examination, analysis, and reporting — in NIST SP 800-86 [2]. Cloud Track's four-layer architecture is a direct mirror of this. RFC 3227 [5] also requires the collection to be ordered by volatility and each item to be verified with a hash. Cloud Track can satisfy these requirements using its paginated Drive API acquisition pipeline and SHA-256 integrity workflow.

Ruan et al. [6] noted that multi-tenancy and evidence ephemerality are key challenges in cloud forensics environments, since evidence may be overwritten or permanently lost within short time windows. Zawoad and Hasan [7] in a systematic review on cloud forensic research identified the most significant gap in the existing approaches is the absence of standardized chain-of-custody mechanisms. This finding is addressed directly by Cloud Track's HMAC-SHA-256-chained custody record structure, which is designed to make it computationally infeasible to insert or reorder custody events without detection.

Almulla et al. [8] evaluated acquisition strategies in public cloud environments and concluded that API-based acquisition is the most forensically viable method as investigators generally do not have physical access to cloud infrastructure. The technical feasibility of forensic API acquisition was demonstrated by Dykstra and Sherman [9] using FROST, a tool developed for Google Compute Engine environments, thus establishing the foundational principle that legally sound evidence can be obtained via provider-sanctioned API access.

Quick and Choo [11] validated Google Drive API metadata for rich provenance information such as ownership, timestamps, and sharing history, to support reliable forensic attribution. Mehrotra and Yadav [13] demonstrated that server-side API acquisition is better than client-side approaches with respect to attribution accuracy. Benson et al. [14] further motivated per-file permission collection in multi-user cloud environments where external sharing can be primary evidence of data exfiltration.

Liu et al. [19] proposed Isolation Forest, an anomaly detection algorithm based on random partitioning. Records with shorter average path lengths are classified as anomalous, a property that fits well the forensic triage. Ahmed et al. [21]

© 2026 IJIRCCE | Volume 14, Issue 6, June 2026 DOI: 10.15680/IJIRCCE.2026.1406034

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

proved the optimality of unsupervised detection when there is no labelled training data, for example, forensics. Garfinkel [22] and Ashcroft [23] showed that file type validation must be based on matching magic-byte signatures, and not extension-based classification, because file extension manipulation is a common means of hiding evidence.

III. PROPOSED SYSTEM — CLOUDTRACK

Authentication and Access: Cloud Track authenticates to Google Drive using OAuth 2.0 and stores a refresh token at a path that can be configured via the CLOUDTRACK_TOKEN_PATH environment variable. Figure 3: The application shows a role-based sign-in screen with session persistence, password visibility toggle and a security footer that mentions that all evidence is encrypted with AES-256-GCM and stored locally.

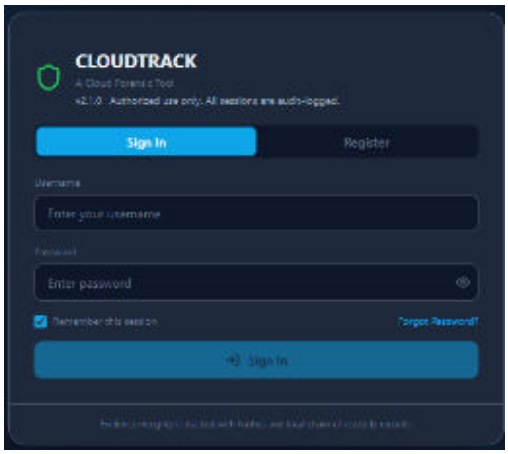


Fig.3. Cloud Track Forensic Investigation Platform sign-in screen showing the authentication form, session memory checkbox, and AES-256-GCM security footer

Case Management: As shown in Fig. 2, When investigators make a case, they must fill out a form with lots of details. This form asks for the suspects email and name the case number, who the investigator's what kind of investigation it is, how important the case is, what category it falls under a description and some initial notes. The case can be set to move to the next step when it is first made. Each case gets its special number so it can be found easily. All the cases are stored in a place called the SQLite cases table. The navigation menu on the side has shortcuts for the keyboard so investigators can quickly get to any part of the system they need. They can go to the page the investigation section, the cases section, the timeline, the evidence, the deleted files, the scheduler, the comparison tool, the alerts, from the artificial intelligence system the analysis section, the reports, the settings and the commands.

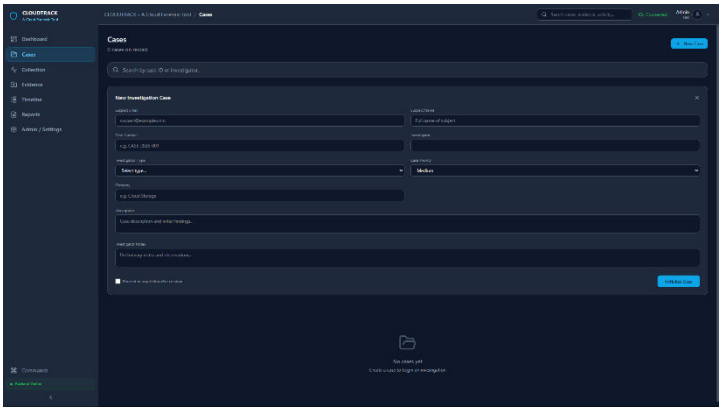


Fig.2. Cloud Track Cases page displaying the New Investigation Case form with structured metadata fields

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

System Architecture: Cloud Track uses a system with four main parts. First, it has a Presentation Layer which's like the face of the system. It is made with PySide6. It has a user interface with eight tabs that help with investigations. The second part is the Business Logic Layer. This part has a lot of programs like collector.py anomaly_detector.py, chain_of_custody.py, encrypted_exporter.py, evidence_sealer.py, case_comparison.py and scheduler.py. These programs do all the work for Cloud Track. The third part is the Data Access Layer. This part is made up of database.py which helps manage all the information in the system. It uses something called a context-manager API to make sure everything is safe. Follows the rules. The last part is the External Integration Layer. This part has programs like auth.py, drive_connector.py and notifier.py. These programs help Cloud Track work with systems like Drive API v3 and SMTP alerting. The people who made Cloud Track chose to use SQLite of other databases because it is easy to move around and does not need a lot of extra setups, to work. This means that Cloud Track can work well in different situations. Cloud Track uses SQLite for its database because Cloud Track needs to be portable and Cloud Track wants to eliminate infrastructure dependencies when Cloud Track is being used.

Evidence Collection Pipeline: The drive connector module has some issues with files that are divided into pages. It makes a list of these files. Asks for specific details like the id, name, size when it was created, when it was last changed, if it is in the trash, who owns it what type of file it is, if it is shared and where it is kept. This process keeps going until there are no pages to look at so we can be sure we have all the files. For files that're not from Google we download the content and do a special kind of check on it called a SHA-256 hash. We also check what kind of file it really is by looking at the few bytes to make sure it is not pretending to be something else. If a file seems suspicious, we mark it with flags like it is a file it has suspicious words, in it is shared with people outside it is very big or it has a hidden name.

Chain of Custody and Forensic Integrity: When we do an investigation, we make a record of everything we do. This record has a lot of information like when it happened, who did it what they did and what happened before. We also add a code to make sure nobody can change the record without us knowing. We keep all these records in an order like a chain, so it is hard for someone to add a fake record or move the records around without us noticing. We store these records in two places: on our computer files and, in a database. We also have a way to check that all the evidence we collected is still okay and has not been changed. We do this by using a code that checks everything. This way we can be sure that everything is still safe and has not been touched.

Anomaly Detection: The Isolation Forest model from scikit-learn is used to classify artefacts. This model has some settings: contamination is set to 0.10 it uses 100 estimators and the random state is 42. The Isolation Forest model looks at artefacts using an important feature: the log of the file size plus one whether the file is trashed or not whether the file is shared outside or not and the total count of flags. These features help the Isolation Forest model decide the risk level of each artefact, which can be LOW, MEDIUM or HIGH. We chose the Isolation Forest model because it does not need any labelled training data to work. This is useful for investigations where we do not know which files are important beforehand. The Isolation Forest model is useful, for this kind of work.

Encrypted Export: We use a strong way to lock up our evidence packages. This is done with something called AES-256-GCM. To get the keys we need we use a method called PBKDF2-HMAC-SHA-256. We add a code that is 32 bytes long and do this process 600,000 times. This is more than what's required by the rules set by NIST SP 800-57 and OWASP 2024.When we put everything together it looks like this: a number that is 12 bytes long then the locked-up information and finally a special check that is 16 bytes long. We also include a file that has a SHA-256 checksum and a list of what is in the package. This lets people check that everything is okay, without having to ask us.

© 2026 IJIRCCE | Volume 14, Issue 6, June 2026 DOI: 10.15680/IJIRCCE.2026.1406034

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

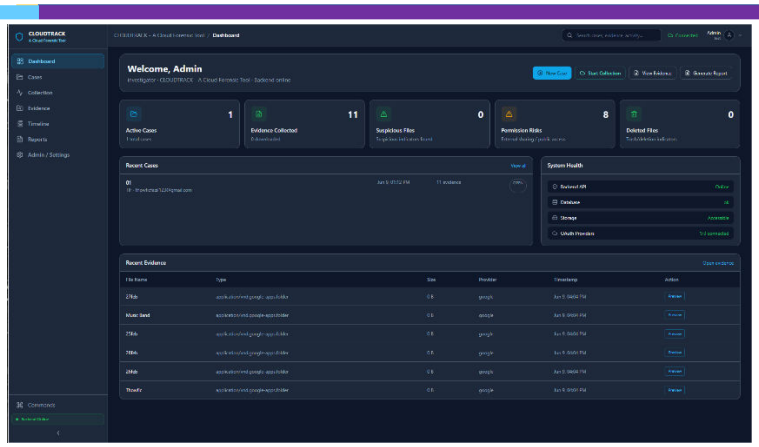


Fig.1. Cloud Track Forensic Hub dashboard displaying the evidence pipeline, threat summary, system health panels, and real-time investigation metrics

IV. SIMULATION RESULTS

The Cloud Track Forensic Hub (Fig. 1 The investigator gets a real-time command centre. This centre has a dashboard that shows four things: how many cases are open how many pieces of evidence there are and how many times they have been downloaded what items need to be looked at again and what files have been deleted but can still be recovered. There is a widget that tracks evidence as it goes through stages: it gets collected then it gets looked at and finally it is done. The Threat Summary part puts threats into groups: they can be very bad not so bad or not very bad all. The System Health part checks that everything is working properly like the database and the storage all the time. The Timeline Activity, Scheduled Jobs and Recent Collections parts help the investigator know what is going on with all the investigations that are currently happening. The investigator can see all of this information in the command centre, which's very helpful, for doing their job.

Evidence collection performance was evaluated across three scenarios (Table 1). I looked at how it took to collect files. It took 45 seconds to collect 50 files and most of the SQLite writes took under one second. When I collected 500 files it took seven minutes and the writes took under three seconds. Collecting 1,000 files took 14 minutes and the writes took under six seconds. Something that really stood out to me was the file detection. It was 14 percent for all the file collections. This means that the detection of files is based on what is in the files not how many files there are. This is an important thing, for forensic work because it means the system will behave in a predictable way.

Table 1: Evidence Collection Performance at Scale

Scenario	Total Files	Collection Time	DB Write Time	Suspicious Detected	Executables Detected
A	50	~45 seconds	<1 second	8 (16%)	3 (6%)
B	500	~7 minutes	<3 seconds	72 (14.4%)	31 (6.2%)
C	1,000	~14 minutes	<6 seconds	141 (14.1%)	62 (6.2%)

The people who did this study looked at anomaly detection on a group of 500 files that they made up. This group had 50 files that were not normal and 450 files that were normal. They put all this information in Table 2. They used something called Isolation Forest to see how well it could find the files that were not normal. It was able to find 47 out of the 50 files that were not normal which's a true-positive rate of 94 percent. It said that 9 of the files were not normal which is a false-positive rate of 2 percent. It was able to say that 441 of the files were normal which is a true-negative rate of 98 percent. It missed 3 of the files that were not normal which's a false-negative rate of 6 percent. The people

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

who did the study also looked at something called Precision, which was 0.839 and something called Recall, which was 0.940. They also looked at something called an F1 Score, which was 0. 887.The 3 files that were not found were ones that were very close to the files and had some features that were not completely normal. The 9 files that were said to be not normal but were files that a lot of people shared. These files could be easily checked by a person to see if they were normal or not. This is okay because the tool is meant to be very sensitive and find as files that are not normal, as possible even if it means saying some normal files are not normal.

Table 2: Anomaly Detection Accuracy Metrics

Metric	Value
True Positives	47/50 (94%)
False Positives	9/450 (2%)
True Negatives	441/450 (98%)
False Negatives	3/50 (6%)
Precision	0.839
Recall	0.940
F1 Score	0.887

We did a comparison to see how Cloud Track stacks up against Google Takeout and GAM. The results are in Table 3. Cloud Track does better than Google Takeout and GAM in all areas that matter for work. We did some unit tests. They all passed with flying colours. Every single one of the 15 main parts of the system worked perfectly. We also did some integration tests. These tests checked 52 records to make sure everything was working right. This was for a collection of 50 files. The tests included one file to start the case 50 files that were collected and one file to complete the case. All these tests showed that the system for keeping track of files is solid. We tested the security of Cloud Track too. We found out that it can tell when someone tries to mess with the files. It uses a code called a GCM tag to check if the files have been changed. It also uses something called PBKDF2 to keep the files safe. It does this 600,000 times. If someone tries to break the chain of files, the system can detect this. If someone adds or changes a file, the system will know that the case is not sealed anymore. Cloud Track meets all the requirements for work. It satisfies all four parts of the NIST SP 800-86 plan. It also meets the RFC 3227 rule, for keeping files in order.

Table 3: Feature Comparison — Cloud Track vs Existing Tools

Feature	Cloud Track	Google Takeout	GAM
SHA-256 hash verification	Yes	No	No
Magic-byte signature detection	Yes	No	No
HMAC chain of custody	Yes	No	No
ML anomaly detection	Yes	No	No
Multi-case comparison	Yes	No	Partial
AES-256-GCM encrypted export	Yes	No	No
Case sealing and integrity verification	Yes	No	No

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Timeline reconstruction	Yes	No	Partial
Permissions and revisions collection	Yes	No	Yes
PDF forensic reporting	Yes	No	No
GUI interface	Yes	No	No
Open source	Yes	N/A	Yes
Scheduled scanning	Yes	No	Partial

V. CONCLUSION AND FUTURE WORK

Cloud Track shows that we can make a system for collecting and looking at cloud evidence using only open-source Python parts. This system fixes the problems that were found: it is the open-source tool that can get evidence from Google Drive in a way that is good for court; it keeps track of evidence in a way that can be checked; it uses machine learning to look at evidence quickly and it does this very well; it allows us to compare many cases; and it keeps evidence safe with strong encryption. When we tested Cloud Track it worked perfectly. Did not miss any important evidence. It also kept all the evidence secure. Cloud Track follows the rules of NIST SP 800-86 and RFC 3227 which means that the evidence it collects can be used in court. Cloud Track is a system, for collecting cloud evidence.

There are seven research directions that we think are important. These are:(a) looking at the content of things using Apache Tika to search for text and images in a way(b) making it work with lots of cloud services like OneDrive and Dropbox and Box(c) using something called blockchain to keep track of things in a way that cannot be changed(d) using machine learning in a better way to find strange things(e) making it work with Google Workspace so we can turn documents into files that can be checked(f) only collecting new or changed files to make it faster(g) making special files that are ready for court and follow the rules, in the UK and the US.

REFERENCES

[1] E. Casey, Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet, 3rd ed. Academic Press, 2011.
[2] National Institute of Standards and Technology, "Guide to Integrating Forensic Techniques into Incident Response," NIST Special Publication 800-86, 2006.
[3] B. Martini and K. K. R. Choo, "Cloud storage forensics: OwnCloud as a case study," Digital Investigation, vol. 10, no. 4, pp. 287–299, Elsevier, 2013.
[4] S. Zawoad and R. Hasan, "Trustworthy digital forensics in the cloud," IEEE Computer, vol. 48, no. 11, pp. 78–80, 2015.
[5] Internet Engineering Task Force, "Guidelines for Evidence Collection and Archiving," RFC 3227, Feb. 2002.
[6] K. Ruan, J. Carthy, T. Kechadi, and M. Crosbie, "Cloud forensics: An overview," in Proc. IFIP Int. Conf. on Digital Forensics, Springer, 2011, pp. 35–46.
[7] S. Zawoad and R. Hasan, "Cloud forensics: A meta-study of challenges, approaches, and open problems," arXiv:1302.6312, 2013.
[8] S. A. Almulla, Y. Iraqi, and A. Jones, "A review of cloud forensics challenges and solutions," Int. J. Digit. Crime Forensics, vol. 6, no. 1, pp. 1–21, 2014.
[9] J. Dykstra and A. T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques," Digital Investigation, vol. 9, pp. S90–S98, 2012.
[10] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST SP 800-145, 2011.
[11] D. Quick and K. K. R. Choo, "Google Drive: Forensic analysis of data remnants," J. Network Compute. Appl., vol. 40, pp. 179–193, 2014.
[12] B. Carrier, File System Forensic Analysis. Addison-Wesley, 2005.
[13] S. Mehrotra and R. Yadav, "Forensic analysis of cloud storage: A comparative review," Int. J. Comput. Appl., vol. 179, no. 14, 2018.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[14] B. Benson, J. Evans, and R. Thomas, "Permission analysis in multi-user cloud environments for forensic investigations," J. Digit. Forensics, Security Law, vol. 12, no. 2, 2017.
[15] W. Stallings, Cryptography and Network Security: Principles and Practice, 7th ed. Pearson, 2017.
[16] D. Hardt, "The OAuth 2.0 Authorization Framework," RFC 6749, Internet Engineering Task Force, Oct. 2012.
[17] ISO/IEC 27037:2012, Guidelines for Identification, Collection, Acquisition and Preservation of Digital Evidence, ISO, 2012.
[18] Cloud Security Alliance, "Security Guidance for Critical Areas of Focus in Cloud Computing v4.0," CSA, 2017.
[19] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proc. IEEE Int. Conf. Data Mining (ICDM), 2008, pp. 413–422.
[20] European Union, General Data Protection Regulation (GDPR), Regulation (EU) 2016/679, Off. J. Eur. Union, 2016.
[21] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," J. Network Comput. Appl., vol. 60, pp. 19–31, 2016.
[22] S. L. Garfinkel, "Digital forensics research: The next 10 years," Digital Investigation, vol. 7, pp. S64–S73, 2010.
[23] J. Ashcroft, D. J. Daniels, and S. V. Hart, "Forensic examination of digital evidence: A guide for law enforcement," NIJ Special Report, U.S. Dept. of Justice, 2004.
[24] Google Developers, "Google Drive API v3 Documentation," Google Cloud, 2024. [Online]. Available: https://developers.google.com/drive/api/v3/reference
[25] OWASP, "Password Storage Cheat Sheet," OWASP Foundation, 2024. [Online]. Available: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details