Forensic Science International: Digital Investigation 44 (2023) 301502

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

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

DFRWS 2023 EU - Selected papers of the Tenth Annual DFRWS Europe Conference

Discovering spoliation of evidence through identifying traces on deleted files in macOS

Jihun Joun, Sangjin Lee, Jungheum Park*

School of Cybersecurity, Korea University, 145 Anam-Ro, Seongbuk-Gu, Seoul, South Korea

ARTICLE INFO

Article history:

Keywords:
Digital forensics
Document forensics
macOS
Spoliation of evidence
e-discovery

ABSTRACT

Spoliation of evidence is a critical concern in various crimes such as information leakage, digital sexual crimes, accounting fraud, and copyright infringement. Several traditional digital forensic investigation methods such as recovery, carving, and anti-forensic behavior tracking are used to investigate these crimes. However, as technology has advanced, recovery and carving have become increasingly challenging. Shortly, data recovery will reach its technological limit, and it will be essential to obtain as much circumstantial evidence as possible based on traces of data left in suspect systems. However, no existing method can systematically track the spoliation of evidence; contemporary investigations typically depend solely on investigators' skills and knowledge. This paper proposes a method to track deleted files by identifying and analyzing various sources that manage file-related metadata in macOS systems.

1. Introduction

Tampering with evidence is fairly widespread in cases of information leakage, accounting fraud, and copyright infringement, as is spoliation in electronic discovery (e-discovery). The preservation of electronically stored information (ESI) is an obligation in criminal investigations (Luoma and Luoma, 2012), (Allman, 2006). The Federal Rules of Civil Procedure 37(e) (2) (C) authorizes a court to impose terminating sanctions, such as a dismissal or a default judgment. Despite strict laws against evidence tampering in most jurisdictions, there have been many deliberate attempts to hinder lawful investigation by law enforcement (Apple Inc. v. Samsung Elecs Co., Ltd, 2012), (Learning Care Group, Inc. v. Armetta, 2016), (Cat3, LLC. v. Black Lineage, Inc., 2016), (BMG Rights Mgmt. (US) LLC. v. Cox Communs., Inc., 2018), (LG Chem, Ltd, et al. v. SK Innovation Co., Ltd, et al., 2019). Consequently, it is essential to establish a claim for spoliation when dealing with certain digital crimes.

Traditional digital forensic methods, such as data recovery (Durrant, 2005), (Garrie, 2014), file carving (Daniel, 2011), and tracing for data wiping tools (Oh et al., 2020), (Conlan et al., 2016) are used to identify data destruction traces. However, these traditional methods have certain limitations. Data wiping software, including operating system (OS) built-in tools such as Disk Utility, can delete certain data and all its traces easily. Given that this software erases the actual data as well as unallocated space, recovery is almost impossible. Even when such tools are not used, if the TRIM function is enabled on a solid-state drive, it permanently deletes free space automatically (Mitchell et al., 2017). Therefore, a standard method is necessary to establish forensically valid proof of deleted file data through traces left on OS and application artifacts.

Currently, investigations in the digital forensic field rely heavily on the experience and know-how of experts (Kafadar, 2019). There is no common standard that sets the qualifications a digital forensic expert must have. Therefore, divergent or inconsistent investigation results may be derived owing to different investigators having different skills and knowledge levels. Without the benefit of a systematic ESI spoliation investigation method, an investigator may submit an analysis report to the court without crucial supporting evidence. This paper provides guidelines to identify potential spoliation of evidence, which can reduce time and workforce requirements and help draw accurate investigation results.

Furthermore, the field of digital forensics has mainly concentrated on Windows systems, posing challenges for examiners who encounter macOS systems during investigations (Schoenhardt, 2020). Digital forensic investigation methods that can be applied to the macOS environment are a crucial requirement. This study aims to find all available traces relating to electronic documents that remain on local storage devices on macOS systems, even after they have been permanently deleted by anti-forensic activities.

* Corresponding author.
E-mail addresses: jihunjoun@korea.ac.kr (J. Joun), sangjin@korea.ac.kr (S. Lee), jungheumpark@korea.ac.kr (J. Park).

https://doi.org/10.1016/j.fsidi.2023.301502

Rather than identifying traces directly related to deleted files, previous digital forensics studies focused on tracking the existence and usage of anti-forensic tools. For example, Rekhis and Boudriga studied a theoretical approach to detect the use of anti-digital forensics tools (Rekhis and Boudriga, 2011). There have also been several attempts at identifying anti-forensic tools' usage traces by using signature detection methods (Park et al., 2017), (Geiger, 2005). Fairbanks et al. developed a journal monitoring tool, TimeKeeper, to extract and analyze anti-forensic attempts. The tool can track file metadata timestamps (modification, access, and change) in honeypots (Fairbanks et al., 2007). However, these methods have certain limitations: well-known filesystem and OS artifacts cannot be maintained over a long term and are often available only for a few days, so the practical applicability of these methods is limited.

We propose a new method to check whether traces of suspected files stored in a target system still exist in the present. Our proposal hypothesizes that it is difficult to erase all traces, and we start with the expectation that traces of deleted files will remain somewhere in the system. To achieve this, we identify and analyze previously unexplored traces that record file-related metadata. In addition, artifacts that are publicly disclosed or known through previous studies are further analyzed from an ESI spoliation perspective to find traces of deleted files. Through these measures, it is possible to identify files that likely once existed but are no longer present.

The main contributions of this paper are: (1) proposing a universal methodology to track deleted files that can be adapted for other files or operating systems, (2) identifying deleted file traces, (3) evaluating the results generated from a self-developed tool, and (4) providing a macOS dataset (containing disk images) created using the proposed methodology.

The remainder of this paper is organized as follows. Section 2 presents the background knowledge and related works. Section 3 explains the intended outcomes of this research. Section 4 describes the adopted methodology. Section 5 explains the analysis findings. In Section 6, we evaluate our methodology and framework through experiments. Finally, we discuss the result and conclude this paper in Section 7.

2. Related works

Spoliation of evidence is a major crime that can have severe legal repercussions. In Black's Law Dictionary, ESI spoliation is defined as the intentional destruction, mutilation, alteration, or concealment of evidence (Black and Garner, 2019). In Arkansas, ESI spoliation is defined as "the intentional destruction of evidence and when it is established, [the] factfinder may draw [an] inference that [the] evidence destroyed was unfavorable to [the] party responsible for its action." (Union Pacific R.R. Co. v. Barber, 2004).

Digital forensic processes, techniques, and tools have been steadily studied to prevent spoliation of evidence, and a legislation related to ESI spoliation has been enacted. Amended rule 37(e) of The Federal Rules of Civil Procedure, which concerns a party's failure to preserve ESI, became effective on Dec. 1, 2015. It considers scenarios wherein, to quote, "electronically stored information that should have been preserved in the anticipation or conduct of litigation is lost because a party failed to take reasonable steps to preserve it." (FRCP Rule 37). Based on this rule, courts can impose severe sanctions for spoliation of evidence, including dismissal, default judgments, or the imposition of attorney's fees.

A 'litigation hold' is a critical phase related to ESI spoliation in the e-discovery process. It is a measure to prevent destruction, deletion, or alteration of all documents and data expected to be of interest from that point on if a lawsuit has already been filed or it is reasonably foreseeable that a lawsuit will be filed. Even if data are not deleted intentionally, it is regarded as an act of obstruction of the trial (Kronisch, 1998), (Silvestri, 2001). Therefore, it is critical to preserve relevant ESI safely.

Bunting studied a specific method for spoliation examination (Bunting, 2016). The author used traces of anti-forensic tools, logs (e.g., disk utility log and system log), quick-look thumbnail cache, trash, and the history of the command list. However, this study was not comprehensive from the perspective of establishing ESI spoliation. Methods specifically aimed at tracing ESI spoliation have not been comprehensively studied; most digital forensic investigators rely on their experience, know-how, and skills to conduct investigations.

Quick and Choo identified types of potential data storage and examined them in relation to data remnants, such as user accounts, passwords, and URLs, on Windows PCs and iPhones. The authors examined the artifacts and network for target cloud storage services, Dropbox (Quick and Choo, 2013a), Microsoft SkyDrive (OneDrive) (Quick and Choo, 2013b), and Google Drive (Quick and Choo, 2014).

In addition to finding traces of deleted files, methods for identifying usage traces of anti-forensic tools have been studied. Geiger discovered significant shortfalls in six anti-forensic tools by observing tools' performance. They found that none of these tools could completely wipe off the unallocated space; residual data could be recovered (Geiger, 2005). Alharbi presented a novel forensic investigation method by analyzing the changes made by anti-forensic tools to the metadata structures of Windows file systems (AlHarbi et al., 2022). Park attempted to identify anti-forensic techniques and tools by using the signature detection method (Park et al., 2017). They used the Indicators of Anti-Forensics (IOAF) tool for automatic anti-forensic trace detection.

In recent years, incidents occurring in Mac environments have increased as Mac computers have become more popular among individuals and business people (Maddu and Rao, 2019). For macOS digital forensic investigation, Casey studied the recent file list (Casey, 2011), and Atwal et al. proposed a new method to identify deleted file lists using Spotlight (Atwal et al., 2019). However, more research is needed to establish ESI spoliation on macOS. We aim to obtain as much circumstantial evidence as possible using the data left in the system in order to establish a claim for ESI spoliation on macOS.

3. Research objectives

As technology has evolved, the traces that have been analyzed thus far to prove spoliation of evidence are inadequate. Newer artifacts and logs that identify traces of deleted files must be discovered. This work verifies previous methods and establishes new analysis methods.

3.1. Spoliation trace

The term "spoliation trace" refers to digital data that indicates the presence of spoliation. Spoliation traces include system and application artifacts where file-related metadata are left even after the relevant file itself is deleted. For instance, spoliation traces can be identified from Windows Prefetch, Eventlog, $LogFile, or even a single application-related log file that records the recently used file's metadata, such as filename, full path, and timestamps.

3.2. Common problems

Current methodologies suffer from the following limitations:

- Limited analysis target - In a case involving spoliation of evidence, a trace of a single deleted file can affect the court decision. Therefore, all artifacts that are likely to have data remnants must be considered during investigations. However, no

comprehensive list of potential sources of spoliation traces has been compiled for investigators' reference.
- Incorrect evidence - Digital forensic investigation must analyze appropriate and accurate evidence. Incorrect information about artifacts and files due to version updates compromises the accuracy of the investigation as false positives and false negatives may occur. Therefore, a systematic framework is needed to follow up new and changed evidence.
- Low reproducibility - There are several available macOS datasets; however, applying them to the experiments to track the trace of files is challenging because the files in the datasets were not systematically created and deleted. In addition, it is challenging to reproduce the same result without a systematic dataset creation method.

3.3. Objectives

The main goal of this work is to develop a systematic methodology for digital forensic investigators to follow. This study has the following four research objectives:

- Proposing a universal methodology to discover all traces of files including unknown artifacts and logs, which can be adapted to any operating system for research, education, and experiments.
- Developing a new macOS dataset (consisting of disk images) for demonstrating our proposal.
- Identifying previously unexplored spoliation traces by examining the developed dataset.
- Evaluating the proposed methodology's usefulness and effectiveness using a self-developed tool.

4. Research methodology

This section describes an overall methodology to identify artifacts that indicate traces of document files. We have composed a dataset based on a variety of detailed user actions. Readers can reproduce the experiments conducted in this study on their storage media and evaluate the validity of the results by following our research methodology. Although our research was limited to a specific OS and several famous document file formats, our methodology is generic and can be applied to other types of OSs and file formats. Fig. 1 shows the flow of our methodology.

4.1. Defining scope

First, the target OS for testing is selected. Forensic analysis methods for Windows OS have been comprehensively studied, and several methods for tracking file traces have been presented. By contrast, compared to its growing popularity, forensic investigation and ESI spoliation examination methods for macOS have rarely been studied. Therefore, this paper establishes a methodology for macOS. Our experiment was conducted on the most recent version, i.e., version 12: Monterey. After picking the target OS, the specific file to be studied is selected. In the e-discovery process, the target ESI is mostly e-documents, such as MS Office documents, e-mails, electronic faxes, TIFF and PDF files, voice records, video, and messages (Schuler, 2011). We focused on e-documents such as MS Office 2007+ documents (Word, Excel, and PowerPoint), PDF, and TextEdit (default macOS text editor).

4.2. Listing the actions

Next, the action list and necessary information to deal with the file are prepared. First, as presented in Table 1, the action list is composed of most common user behaviors such as ‘open’, ‘save’, or ‘move’. Then, the filename and content that will be used in the experiment must be created. As the filename and content will be used for keyword search when discovering traces later, unique filenames and content should be used in order to avoid false positives. Table 1 targeted only the action list for the MS Word file format as an example; other file types can be considered for further experiments.

4.2.1. File creation

This study employs TextEdit (the default word processor in macOS), MS Word, and PDF files. The action list includes creating a new file, saving a file to another file format, creating an alias file, and duplicating a file.

4.2.2. File access

There are several methods to access the files. First, most users open the files by double-clicking the target file or clicking each application's ‘open’ button. Second, a file can be opened on Spotlight, the macOS search engine, by searching for the filename. Finally, files sent, received, or saved by messenger or cloud storage can be opened on that application without downloading. In our experiment, we targeted Microsoft Teams, Slack messenger, Google Drive (local folder and Web browser), and iCloud. We also intentionally damaged a file to check what data would be written to the log file in case of such an error.

4.2.3. File modification

This section describes how to modify files. Two properties of a file can be modified by the user: the filename and the content. In our experiment, we renamed a file to see if we could find any association between the old name and the new name after renaming. Then, there are three action types that modify content: adding content, deleting partial content, and deleting entire content. We also included embedded files to ensure that the metadata of embedded files or objects exists after deletion.

4.2.4. File copy and up/download

We considered three actions related to moving and copying: from external drive to local drive, local drive to external drive, and between the local volumes. This section describes methods to upload and download the files online. Cloud storage (Google Drive and iCloud), messenger application (Microsoft Teams - web browser and application), mail (Gmail), and web browser (Safari) were considered for these actions.

4.2.5. Others

The Others section lists actions other than those listed above, including synchronization, sharing, compression, and printing. iCloud and Google Drive are used to synchronize the files. ‘Sharing’ is a function supported by Apple, wherein files can be shared via Mail, Message, AirDrop, and Notes.

4.3. Creating the dataset

Based on the actions defined above, the dataset consists of four steps: performing the action, imaging the volume, deleting files, and imaging the disk volume again. First, the actions are performed on the system in a clean state. Second, data is collected via two methods: imaging the disks or creating snapshots using a virtual environment. Either method can be used, but imaging provides more accurate results because unexpected outputs can be found in the virtual environment. Next, the actual files created are permanently deleted. Finally, the disks are imaged again and the result is compared with the previous image file. For reference, we imaged

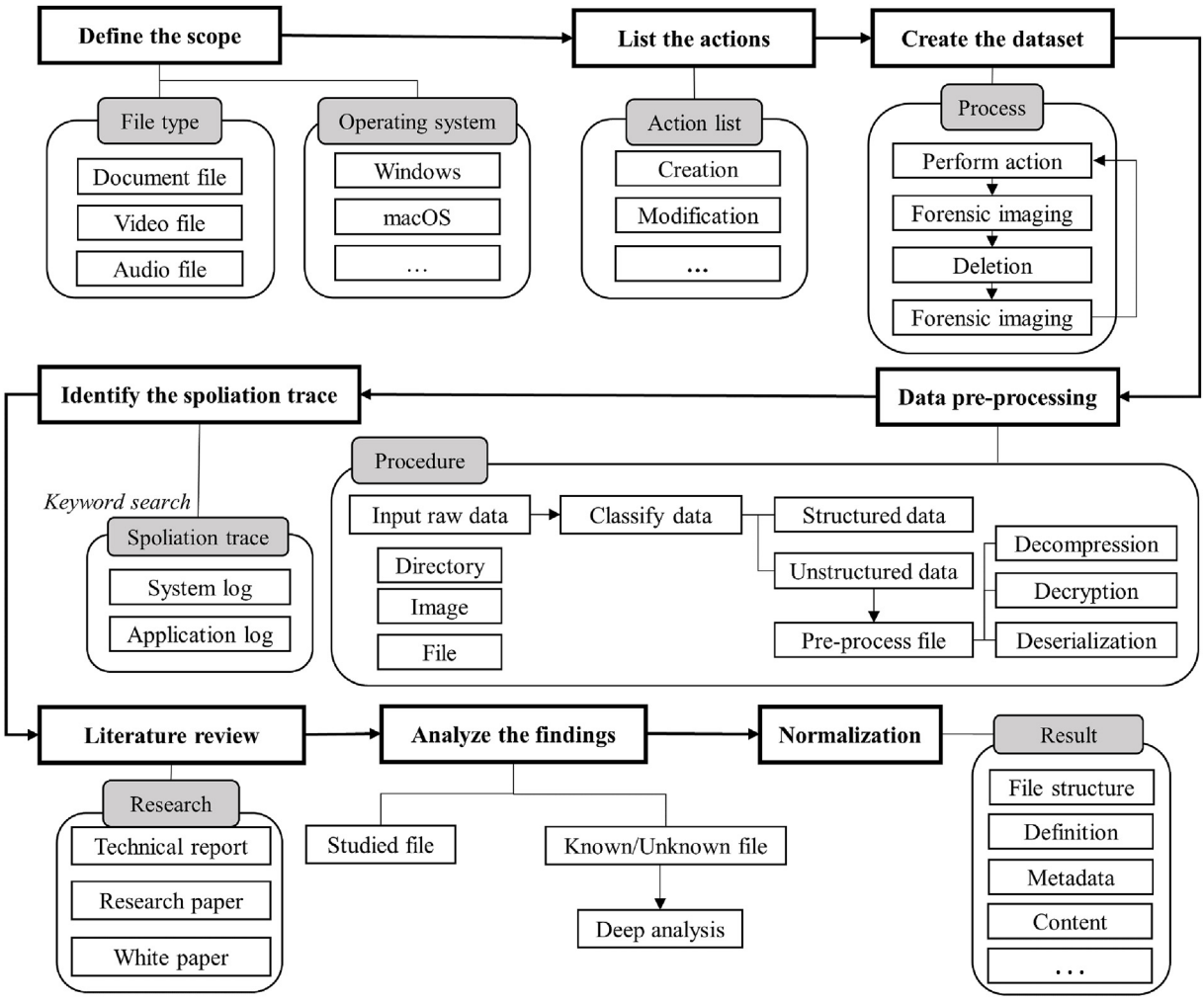


Fig. 1. Research methodology diagram to identify trace evidences related to deleted documents.

Table 1
Action list and description.

Action	Description
File Creation	Creating new files
File Access	Opening and reading the files
File Modification	Modifying and deleting the files
File Copy	Copying and moving the files
File Up/Download	Up/downloading the files from online
Others	Other operations on the files

and focused on the allocated area only. Unallocated space is not our target area because recovery or carving is not the aim of this study.

4.4. Data pre-processing

After this phase, traces should be identified through keyword search targeting all files in the dataset. Thus, this data pre-processing is performed to create a searchable environment. The input source will be the image file created by forensically imaging the dataset or file/directory extracted from the dataset. Next, the examiner should check whether the identified file is compressed, serialized, or encoded. The file content and metadata can be identified through decompression, decryption, and deserialization. However, if the file format is unknown or encrypted, the contents should be verified through additional research and analysis. As most files have been previously studied by numerous researchers, analyzing them is straightforward. For instance, bulk data analysis tool (Garfinkel, 2013) can be helpful to complete this step.

4.5. Identifying the spoliation trace

In this phase, all spoliation traces (artifacts and logs) will be identified by performing searches using filenames and contents as keywords. We also search by the renamed filename, modified content, and deleted content. Depending on file format, additional files should be collected and analyzed. For instance, SQLite format files always record the transaction log in the 'wal' file. This file contains valuable information that does not exist in the current database file (Liu et al., 2016). Thus, for files in SQLite format, the 'wal' file must be analyzed to find traces of deleted files. The output of this step will be the list of artifacts and logs that stores entire or partial document-relevant data, e.g., Spotlight logs, recent file logs, and MS Office logs.

4.6. Literature review

This stage involves reviewing the list found in the previous stage and dividing it into studied traces and known/unknown traces. It is essential to review previous literature to check whether the identified files have already been studied. Technical reports, research

papers, white papers, and conference papers are reviewed accordingly. Additional analysis is not necessary if the identified artifacts and logs are previously known. However, undiscovered files should be analyzed to determine which metadata related to the deleted file is recorded.

4.7. Analyzing the findings

Next, we analyze the list of traces to find document-relevant data. For instance, if the examiner finds a log file in SQLite format through a keyword search, they analyze what document file-related traces exist in the file. For studied traces, use existing forensic tools or scripts to identify and list data remnants inside the traces. And for known/unknown traces, interpret internal storage structures to identify and list data remnants inside the traces through additional research and development.

4.8. Normalization

As the information and the format stored are different for each spoliation trace, it is essential to organize them through normalization. In this phase, we list and normalize the traces related to files in which spoliation traces exist like Table 2.

5. Spoliation trace analysis for digital forensics

We identified and analyzed traces of deleted files through our proposed methodology. Some files' formats and definitions have been described in previous studies and official documents. However, it is challenging to establish ESI spoliation based on previous studies only. In particular, there is no research on whether traces are left in internal artifacts even after the actual files are permanently deleted.

Table 2 summarizes the analysis result and list of the identified spoliation traces. The metadata column shows the types of metadata that can be found in artifacts. We defined types of metadata using six features: Filename (N), path (P), timestamp (T), size (S), content (C), and the file itself (I). We also divided the specified files into three categories (Unknown, Known, and Studied traces).

- Unknown traces: Files that are identified for the first time in this study.
- Known traces: Files that are known already, but the file traces that they may remain have not been studied.
- Studied traces: Files that are known already and the file traces that they may remain have already been studied.

We analyzed all identified spoliation traces but focused particularly on 'Unknown traces', which is a new and meaningful trace. Analysis of 'Known traces' and 'Studied traces' with more detailed descriptions, are available on GitHub.[1]

5.1. Operating system - spotlight

There are numerous valuable data in Spotlight on macOS. Spotlight is a desktop search technology first released with Mac OS X 10.4 (Tiger) and is included in recent macOS versions. It has already been studied for digital forensic analysis. In particular, the "store.db" file is well known, but other related files have not yet been analyzed. We discovered traces of deleted files in parsecd and journalAttr files.

"Parsecd" is a temp log file used for Spotlight, messages, lookup, and Safari. This file's information and structure are not accessible to the public and have not been analyzed. The filename is saved with names such as "session.[random].open" or "session.[random].-close" in "~/Library/Caches/com.apple.parsecd/". As shown in Fig. 2a, we discovered that the deleted filename with or without its extension is stored in this file.

"JournalAttr" is a journal file for the Spotlight indexing function, and it has not been previously analyzed. This file is saved in "~/Library/Metadata/CoreSpotlight/index.spotlightV3" and "/.Spotlight-V100/Store-V2/[random]". Fig. 2b shows that deleted filename and metadata are stored temporarily when the file is uploaded or downloaded from the iCloud.

"Store.db" file contains the details of Spotlight searches have already been analyzed by many researchers. The database contains the file system characteristics, metadata, and indexed textual content that helps us to find files and information (Apple, 2013). The deleted filename, path, size, and timestamps are stored in this file (Khatri, 2019). Atwal studied the persistence of records for deleted files in the storage for Spotlight and checked whether the deleted database pages are recoverable from unallocated space (Atwal et al., 2019). Khatri analyzed the structure of the Spotlight metadata cache database and developed a script to explore and read the database (Khatri, 2019).

5.2. Document application artifacts - microsoft office

Document application creates specific logs and files to manage the document files. These can be used to trace the deleted file. In this study, MS Word is the target document application.

"ComRPCDB" is a database file that has not been analyzed. This file is saved in "~/Library/Group Containers/UBF8T346G9.Office/ComRPC32". The database records the information of the embedded object within a document file. The filename and path are stored from the offset '0x10' in the 'moniker_eq_buff' column of the 'rot' table in the database. In the 'proc_time' column of the same table, the creation time of the embedded file is stored as a Unix timestamp. As it is an SQLite database, traces of deleted files can be found in the "ComRPCDB-wal" file too.

"MicrosoftRegistrationDB_[random].reg" is a database file that has not been analyzed. This database contains information about files created or downloaded directly. The file is stored in "~/Library/Group Containers/UBF8T346G9.Office/MicrosoftRegistrationDB". The table "HKEY_CURRENT_USER_values" contains the filename, path, and the timestamp. Each file is separated by the value of the 'node_id' column. The data types of the 'name' columns are 'DocumentURL', 'FileName', 'FileSizeinBytes', 'FutureAccessToken', 'isPinned', 'Path', 'Timestamp', and so on. The actual metadata are stored in the 'value' columns in the same table.

The file "com.microsoft.Word.plist" is a plist file that stores various information such as application version at boot time, installed language, and last accessed folder name. It was experimentally confirmed that the deleted filename and path could be identified in this database file. The key with a name that starts with 'NSWindows Frame' records contains the accessed filename and path. This file is located in " ~/Library/Containers/com.microsoft.-Word/Data/Library/Preferences".

The file "com.microsoft.Word.securebookmarks.plist" is a well-known log file in MS Word. This file exists in the "~/Library/Containers/com.microsoft.Word/Data/Library/Preferences" folder. The MS Office application displays the recent file list by referring to this file. This file has already been analyzed; recently accessed files' names, paths, and timestamps are saved in this file.

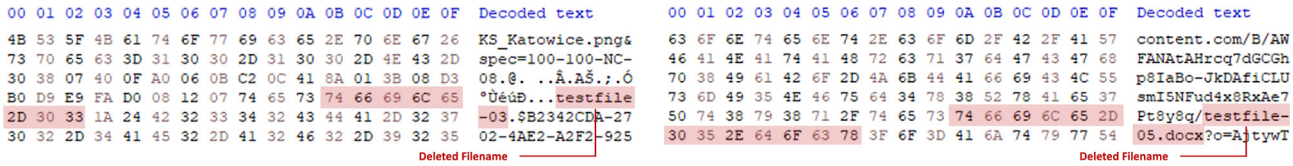
5.3. Cloud storage artifacts - Google Drive

E-mail, external storage devices, messengers, and cloud storage

[1] https://github.com/blackmax90/TraceEvidence-on-macOS.

Table 2
Result of the spoliation trace analysis for digital forensic investigation in macOS.

Type	Artifact	Novelty	Filename	Metadata					
				N	P	T	S	C	I
OS	Spotlight	Known	com.apple.spotlight.Shortcuts	O	O	O			
		Unknown	Parsecd	O					
		Unknown	JournalAttr.[filename]	O					
		Studied	Store.db	O	O	O			
	Diagnostic	Known	[random].tracev3	O	O	O			
		Known	[year].[month].[day].asl	O		O			
	Recent Files	Studied	com.apple.textedit.sfl2	O	O				
		Studied	~RecentDocuments.sfl2	O	O				
	Document Revision	Studied	db.sqlite	O		O			
	DS Store	Studied	.DS_Store	O	O		O		
	FSEvent	Studied	.fseventsd	O	O	O			
	KnowledgeC	Known	knowledgeC.db	O					
Document	MS Office	Unknown	com.microsoft.Word.plist	O	O				
		Unknown	ComRPCDB	O	O	O			
		Unknown	MicrosoftRegistractionDB_.reg	O	O	O	O		
		Studied	~securebookmarks.plist	O	O	O			
	Notes	Studied	[filename]	O	O	O	O	O	O
		Known	NoteStore.sqlite	O		O			
Cloud Storage	Google Drive	Unknown	(mirror) metadata_sqlite_db	O		O	O		
		Unknown	structured_log_[random]	O					
		Unknown	finder_ext_[random].txt	O	O	O			
		Unknown	drive_fs_[random].txt	O	O	O			
	iCloud	Studied	[filename]	O	O	O	O	O	O
		Known	client.db, server.db	O		O	O		
Messenger	Microsoft Teams	Known	[random]_0	O	O	O	O	O	O
		Known	[random].log	O					
Web	Safari	Studied	Downloads.plist	O		O			
		Known	[random], [random]-blob	O					
		Studied	History.db	O					
		Studied	Favicons.db	O					
		Studied	RecentlyClosedTabs.plist	O		O			
Mail	Mail (App)	Known	Envelope Index	O		O			



(a) Parsecd (b) JournalAttr

Fig. 2. Unknown spoliation trace from "Spotlight" that contains the filename.

are the primary target of ESI spoliation investigation. This study aims to identify traces of files left in local storage when using Google Drive. Google Drive creates databases and logs to manage the files in cloud storage. These databases and logs can be used for digital forensic investigations.

The file stored in Google Drive is recorded on "metadata_sqlite_db" and "mirror_metadata_sqlite.db" files. Once the file is stored in Google Drive, the file information is recorded in these databases. The "metadata_sqlite_db" file stores a list of all files and folders that currently exist. The "mirror_metadata_sqlite.db" database has the same structure as "metadata_sqlite_db," and it can be considered a file of the journal function. Both files can be found in "~/Library/Application Support/GoogleDriveFS/random". The filenames, paths, and sizes are stored in several tables in these two databases. In the 'items' table, the filename is stored in the 'local_title' column, and the size is stored in the 'file_size' column. The last modified time is stored as 'modified_date', and the last accessed time by the user is stored as 'viewed_by_me_date'. The filename is stored in another location in this table, i.e., the 'proto' column. The filename also exists in the value in the 'item_properties' table. It is stored in the 'value' column in the row where the 'key' column value is 'local-title'. Finally, in the 'proto' column in the 'deleted_items' table, information related to the document file that has been deleted and moved to the recycle bin is stored. The "structured_log_[random]" and "finder_ext_[random].txt" files are logs generated by Google Drive, which have not yet been disclosed or studied. These are in the "~/Library/Application Support/Google/DriveFS/Logs" folder. "Structured_log_[random]" file contains the filename and extension of all files uploaded or downloaded from the Google Drive web, synchronized in a local synchronization folder, viewed, modified, or saved. The "finder_ext_[random].txt" is expected to record logs of errors and warnings that occur when accessing files in Google Drive. The accessed filename, path, and timestamp are stored in this file.

6. Experiments and evaluation

In this section, we describe the datasets we created using the proposed methodology and the result. We evaluated the accuracy by identifying the number of deleted files. The dataset created for

the experiments is shared from a Google Drive link on GitHub. In addition, all data and results used in the experiments have been uploaded in order to allow anyone to validate our methodology and results. Fig. 3 is an experiments design diagram that shows our experimental procedure.

6.1. Experiments

We produced two experimental laptop images in which 54 document files were created and deleted. One is designed as used by an ordinary user, and the other is designed as used by a digital forensic expert with knowledge of the artifacts studied and discovered thus far. Materials and procedures used to create the dataset were uploaded on GitHub. Owing to privacy issues, traces left by operations of file sharing via iMessage and printing were not included in these datasets.

6.1.1. Advance preparations

Before starting the experiments, the necessary items were prepared. First, we erased all content and restored it to factory settings. Then, we decided on a list of 54 actions to perform, presented in Table 3. In order to find traces of deleted files using a keyword search without false positives, a unique filename, path, and content were used for each file.

6.1.2. Ordinary user image

"Ordinary user image" is a forensic disk image created as if an ordinary user had destroyed the evidence while using the laptop. All actual files created with the filename of 'randomfilename[number]' and related files were permanently deleted in both the laptops and the applications. We used two operations to delete recent traces easily: 'Menu (Apple logo) - Recent items - Clear menu' and 'MS Word - File - Open recent - More - Remove from recent'. In addition, "~/Recentdocument.sfl2" and "~/securebookmarks.plist", the most well-known files related to the recent file, were deleted.

6.1.3. Specialist image

"Specialist image" is the image created when a malefactor proficient in knowledge related to evidence destruction uses the laptop. In addition to the "Ordinary user image", we removed all artifacts and logs that are 'Known traces' and 'Studied traces'. As mentioned in Section 5, 'Known traces' is an artifact or file previously discovered and defined. However, the 'Known traces' has not yet been studied from the perspective of spoliation of evidence, so the possibility of using them for spoliation investigation was discovered for the first time in this study. We created this image to determine whether our proposed method can be used even when the suspect has erased all traces that are difficult for even a digital forensic expert to recognize.

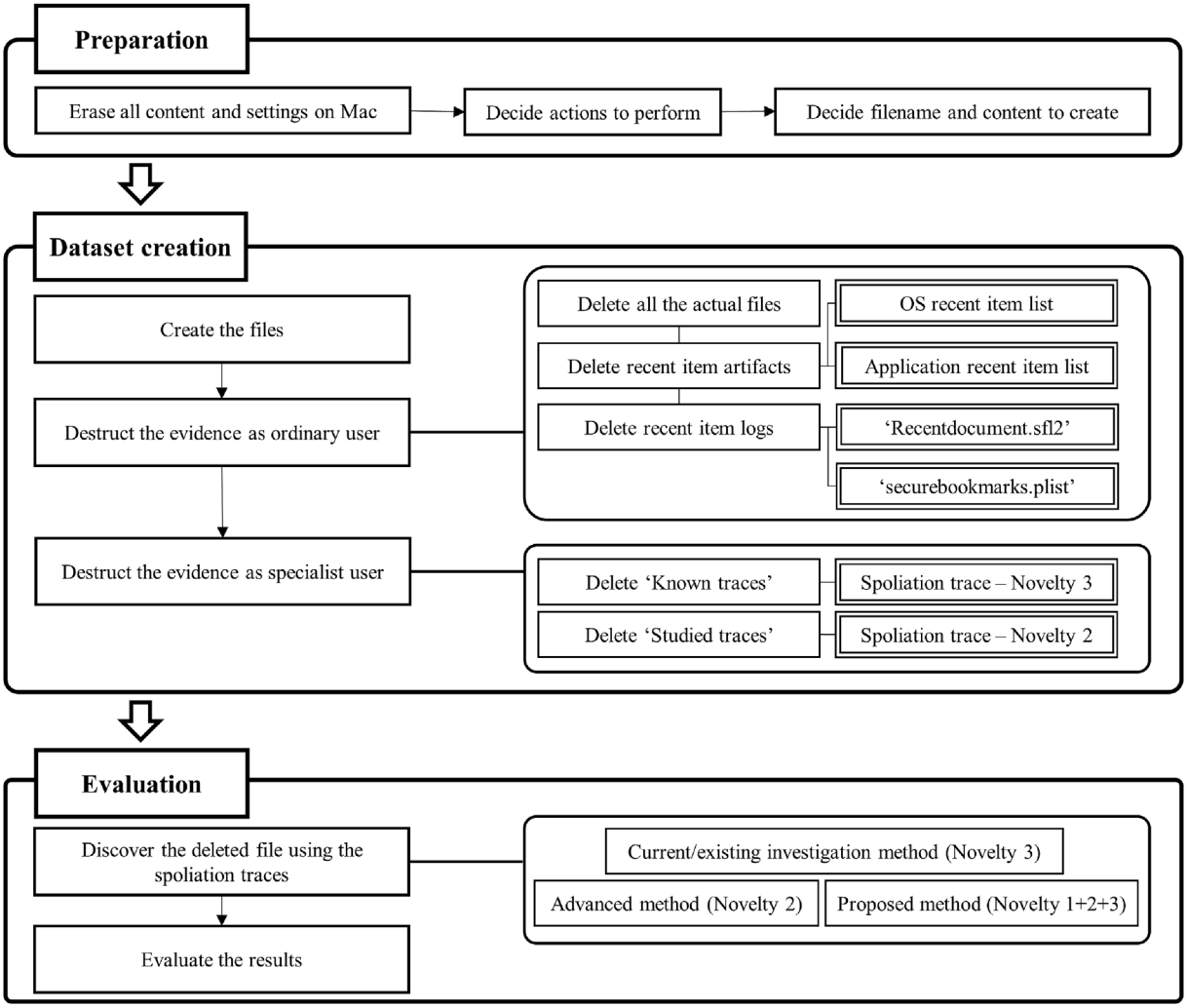


Fig. 3. Experiments design diagram.

Table 3
Experiments summary table.

	Action	Filename	Unknown	Known	Studied	Total
Create	Create file (TextEdit)	randomfilename01.docx			O	O
	Create file (MS Word)	randomfilename02.docx	O			O
	Save as	randomfilename03.docx	O			O
Alias	Alias	randomfilename04.docx	O			O
		randomfilename04 alias.docx				
Duplicate	Duplicate	randomfilename05.docx	O			O
		randomfilename05 copy.docx				
Share	AirDrop	randomfilename06.docx				
	Mail	randomfilename07.docx		O		O
	Notes	randomfilename08.docx	O		O	O
Compress	Compress	randomfilename09.docx				
		randomfilename10.docx				O
		Archive.zip				
Copy	Ext -> Int	randomfilename11.docx				
	Int -> Int	randomfilename12.docx	O			O
	Int -> Ext	randomfilename13.docx	O			O
Move	Ext -> Int	randomfilename14.docx				
	Int -> Int	randomfilename15.docx	O			O
	Int -> Ext	randomfilename16.docx	O			O
Synchronize	Google Drive (App)	randomfilename17.docx	O			O
		randomfilename18.docx	O			O
	iCloud (App)	randomfilename19.docx	O	O	O	O
		randomfilename20.docx	O	O	O	O
		randomfilename21.docx	O	O	O	O
Download	Teams (App)	randomfilename22.docx		O		O
	Google Drive (Web)	randomfilename23.docx	O		O	O
	iCloud (Web)	randomfilename24.docx	O	O	O	O
	Gmail (Web)	randomfilename25.docx		O	O	O
	Internet	randomfilename26.docx	O	O	O	O
Upload	Teams (App)	randomfilename27.docx		O		O
	Google Drive (Web)	randomfilename28.docx	O			O
	iCloud (Web)	randomfilename29.docx		O		O
	Gmail (Web)	randomfilename30.docx	O	O		O
Open	Double click	randomfilename31.docx	O			O
	Spotlight	randomfilename32.docx	O		O	O
	Properties	randomfilename33.docx			O	O
	Damaged file (Click)	randomfilename34.docx	O			O
	Damaged file (Spot~)	randomfilename35.docx	O			O
	Teams (Open in web)	randomfilename36.docx	O	O	O	O
	Teams (Open in app)	randomfilename37.docx	O	O	O	O
	Slack	randomfilename38.docx				O
	iCloud (App)	randomfilename39.docx	O	O		O
Modify	Rename	randomfilename40.docx				
		randomfilename41.docx	O	O	O	O
	Add content (end)	randomfilename42.docx	O			O
	Add content (middle)	randomfilename43.docx	O			O
	Modify content	randomfilename44.docx	O			O
	Teams (Web)	randomfilename45.docx	O	O	O	O
	Teams (App)	randomfilename46.docx	O	O	O	O
	iCloud (App)	randomfilename47.docx	O	O		O
	Delete partial content	randomfilename48.docx	O			O
	Delete all content	randomfilename49.docx	O			O
	Add embedded file	randomfilename50.docx	O			O
		randomfilename51.docx	O			O

Table 4
Number of deleted files identified by each method.

Methods	Current method (Novelty 3)	Advanced method (Novelty 2)	Proposed method (Novelty 1+2+3)
Ordinary User	16/54	22/54	42/54
Specialist User	0/54	0/54	32/54

6.2. Evaluation

No framework or guidelines for investigating spoliation have been studied yet, making it difficult to compare the accuracy of our method. Therefore, we judged which spoliation trace files could be identified and checked how many files could be identified by our proposed method in the images. However, traces related to 'fsevents' were excluded as this artifact only stores records for a short period depending on system usage. Table 3 categorizes the files that can be identified.

We developed a tool using Python 3.10 to automate the gathering of deleted file traces from running machine as well as files extracted from a forensic image as an input source. The tool collects all the trace evidence and lists all files may have existed. Thus, it can be used to find traces of deleted files after comparison with existing files. We used this tool to evaluate our proposed methodology and have uploaded our code to GitHub.

We defined three methods and compared the number of identified traces of deleted files in the two images. As presented in Table 4, we confirmed that more files were identified in all images compared to the existing methods. 'Current method' is the existing spoliation of evidence investigation method using 'Studied traces'. In the ordinary user image, 16 traces were identified using the 'Current method'. As this image represents a laptop on which an ordinary user tried to destroy evidence, some traces could be identified using the conventional method. However, in the specialist image, nothing was found. The second method, 'Advanced method,' uses the spoliation traces defined as 'Known traces.' In the ordinary user image, 22 traces were identified, and higher accuracy than the conventional analysis method was confirmed. However, no trace was found in the specialist image. Our proposed methods (using Unknown, Known, and Studied traces) demonstrated the best performance. In the ordinary user image, 42 files were identified, and in the specialist image, 32 files were identified. Thus, the performance of the proposed identification method was verified.

The spoliation trace for each action has been uploaded on GitHub. Through this, it is possible to know which trace to look for in each behavior, which can be used for future research. These can be verified using the data set and answer sheet published on GitHub, and these can be used for future studies related to macOS and document files.

7. Discussion and conclusions

This study proposes a novel method to establish ESI spoliation. It identifies verifiable evidence using minimal data, even when the file has been completely deleted. In addition, it establishes a systematic and standard methodology that all investigators can use to derive the same results without false positives for finding traces of ESI spoliation. Using the proposed methodology, new related traces can be identified even in files other than document files or in the operating system.

We first determined all actions that a user performs on files. Then, the file traces left by each of the actions were identified. There may be actions other than the listed actions, and traces may not appear depending on the OS or application version. However, the artifacts and log files for those actions can be identified by experimenting with the old or new version of the OS or application through the established systematic methodology.

Our study has certain limitations. Some artifacts and log files leave only the filename. To check whether the file is the same, the investigator has to distinguish it through a hash value or other metadata, but it cannot be confirmed that it is the target file only on the basis of the filename. However, in e-discovery with strong sanctions such as litigation hold, concerns arise whether a file is intentionally or accidentally modified or damaged. Therefore, if the actual file does not exist, it can be argued that it was modified even though the only trace that can be found is the filename.

In this study, several artifacts, files, and logs that leave traces of document files are identified, and these traces can be found using the keyword search. However, there are some deficiencies in the identified file structure and analysis method. Therefore, we plan to analyze the various artifacts found by us in-depth to check whether the artifact has alternative digital forensic meanings.

We also plan to look for a trace of evidence of destruction in the Windows OS. The artifacts that exist in the Windows OS have been studied by many experts but, as with macOS, very few studies specifically consider ESI spoliation. Therefore, in the future, we plan to identify and analyze all artifacts, files, and logs that leave traces of document files in the Windows operating system.

We have identified various new artifacts and logs that may have forensic significance in macOS. It is possible to prove ESI spoliation based on our findings. These analysis results can contribute to real-world investigative situations as the results were tested using actual data created using the methodology. Our systematic methodology can be used to find traces on a variety of OSs and files, such as other word processors (iWork and Libre Office), e-mail, or voice records. As such, investigators with differing levels of knowledge and skills can obtain the same results, which can help forensic investigations.

Acknowledgements

This work was supported by Police-Lab 2.0 Program(www.kipot.or.kr) funded by the Ministry of Science and ICT(MSIT, Korea) & Korean National Police Agency(KNPA, Korea). [Project Name: Research on Data Acquisition and Analysis for Counter Anti-Forensics/Project Number: 210121M07].

References

AlHarbi, R., AlZahrani, A., Bhat, W.A., 2022. Forensic analysis of anti-forensic file-wiping tools on windows. J. Forensic Sci. 67 (2), 562–587.
Allman, T.Y., 2006. Managing preservation obligations after the 2006 federal e-discovery amendments, Rich. J. Technol. 13, 1.
Atwal, T.S., Scanlon, M., Le-Khac, N.-A., 2019. Shining a light on spotlight: leveraging apple's desktop search utility to recover deleted file metadata on macos. Digit. Invest. 28, S105–S115.
Black, H.C., Garner, B.A., 2019. Black's Law Dictionary, eleventh ed. St. Paul, MN.
Bunting, S., 2016. Forensic analysis of spoliation and other discovery violations part 1: macintosh examinations. eForensics Magazine.
Casey, E., 2011. Digital evidence on macintosh systems. Digital Evidence and Computer Crime 587.
Conlan, K., Baggili, I., Breitinger, F., 2016. Anti-forensics: furthering digital forensic science through a new extended, granular taxonomy. Digit. Invest. 18, S66–S75.
Daniel, L., 2011. Digital Forensics for Legal Professionals: Understanding Digital Evidence from the Warrant to the Courtroom. Elsevier.
Durrant, R., 2005. VII. Spoliation of discoverable electronic evidence. Loyola Los Ang. Law Rev. 38 (4), 1803.
Fairbanks, K.D., Lee, C.P., Xia, Y.H., Owen, H.L., 2007. Timekeeper: A metadata archiving method for honeypot forensics. In: 2007 IEEE SMC Information Assurance and Security Workshop. IEEE, pp. 114–118.
Garfinkel, S.L., 2013. Digital media triage with bulk data analysis and bulk extractor. Comput. Secur. 32, 56–72.
Garrie, D.B., 2014. Digital forensic evidence in the courtroom: understanding content and quality, Nw. J. Tech. & Intell. Prop. 12, i.
Geiger, M., 2005. Evaluating Commercial Counter-forensic Tools. 2005 Digital Forensic Research Workshop (DFRWS).
Kafadar, K., 2019. The need for objective measures in forensic evidence. Significance 16 (2), 16–20.
Khatri, Y., 2019. Investigating spotlight internals to extract metadata. Digit. Invest. 28, 96–103.
Kronisch, v. United states, 1998,, p. 126 (2d cir. 1998).
Liu, Y., Xu, M., Xu, J., Zheng, N., Lin, X., 2016. Sqlite forensic analysis based on wal. In: International Conference on Security and Privacy in Communication Systems. Springer, pp. 557–574.
Luoma, M., Luoma, V., 2012. After five years of e-discovery missteps: sanctions or safe harbor? Annual ADFSL Conference on Digital Forensics, Security and Law.
Maddu, B., Rao, P., 2019. Os x artifact analysis. Int. J. Recent Technol. Eng. 7, 26–32.
Mitchell, I., Anandaraja, T., Hara, S., Hadzhinenov, G., Neilson, D., 2017. Deconstruct and preserve (dap): a method for the preservation of digital evidence on solid state drives (ssd). In: International Conference on Global Security, Safety, and Sustainability. Springer, pp. 3–11.
Oh, D.B., Park, K.H., Kim, H.K., 2020. De-wipimization: Detection of data wiping traces for investigating ntfs file system. Comput. Secur. 99, 102034.
Park, K.J., Park, J.-M., Kim, E.-j., Cheon, C.G., James, J.I., 2017. Anti-forensic trace detection in digital forensic triage investigations. Journal of Digital Forensics, Security and Law 12 (1), 8.
Quick, D., Choo, K.-K.R., 2013a. Dropbox analysis: data remnants on user machines. Digit. Invest. 10 (1), 3–18.

Quick, D., Choo, K.-K.R., 2013b. Digital droplets: microsoft skydrive forensic data remnants. Future Generat. Comput. Syst. 29 (6), 1378–1394.
Quick, D., Choo, K.-K.R., 2014. Google Drive: Forensic analysis of data remnants. J. Netw. Comput. Appl. 40, 179–193.
Rekhis, S., Boudriga, N., 2011. A system for formal digital forensic investigation aware of anti-forensic attacks. IEEE Trans. Inf. Forensics Secur. 7 (2), 635–650.
Schoenhardt, S., 2020. Macintosh Apfs Forensic Software Assessment: Blackbag Technologies Blacklight 2019 R3. Ph.D. thesis. Utica College.
Schuler, K.A., 2011. E-Discovery: Creating and Managing an Enterprisewide Program: A Technical Guide to Digital Investigation and Litigation Support. Syngress.
Learning Care Group, Inc. v. Armetta, 2016, p.434 (D. Conn. 2016).
Silvestri v. Gen Motors Co., 2001, 271 f.3d 583 p.591 (4th cir. 2001).
The Federal Rules of Civil Procedure Rule 37, 2015.
Cat3, LLC. v. Black Lineage, Inc., 2016, F.Supp, p.488 S.D.N.Y.
BMG Rights Mgmt. (US) LLC. v. Cox Communs., Inc., 2018, 881 F.3d 293 (4th Cir. 2018).
Apple, 2013, Spotlight overview. URL https://developer.apple.com/library/archive/documentation/Carbon/Conceptual/MetadataIntro/MetadataIntro.html.
Union Pacific R.R. Co. v. Barber, 2004, 356 ark. 268, 298, 149 S.W.3d 325, 345.
Apple Inc. v. Samsung Elecs Co., Ltd., 2012, 888 F. Supp. 2d 976, 989 (N.D. Cal. Aug. 21, 2012).
LG Chem, Ltd, et al. v. SK Innovation Co., Ltd, et al., 2019, Inv. No. 337-TA-1159.