Designing and Validating an Evaluation Digital Forensic for Selective Seizure Capabilities in Windows Forensic Tools

Sun-Ho Kim and Cheolhee Yoon *

Posted Date: 22 January 2026

doi: 10.20944/preprints202601.1696.v1

Keywords: digital evidence; digital forensic investigation; evaluation model; NTFS parsing; selective seizure functions; windows forensic tools; windows log analysis

Article

Designing and Validating an Evaluation Digital Forensic for Selective Seizure Capabilities in Windows Forensic Tools

Sun-Ho Kim [1] and Cheolhee Yoon [2,*]

[1] Ablesecu, Seoul. Republic of Korea

[2] Laboratory of Autonomous Vehicle and Block-chain, Korean National Police University, Republic of Korea

* Correspondence: bertter@police.ac.kr

Abstract

The increasing volume and complexity of digital evidence pose significant challenges to its lawful collection and admissibility, particularly in on-site investigative contexts. Selective seizure has emerged as a critical approach for minimizing unnecessary data acquisition while ensuring procedural legality, privacy protection, and investigative efficiency. However, despite its growing importance, systematic evaluation criteria for selective seizure capabilities in digital forensic tools remain underdeveloped. This study proposes a structured evaluation framework for assessing selective seizure functions in Windows-based forensic tools, with a focus on live-response environments. Essential selective seizure functions were identified and organized into three investigative phases—search, selection, and seizure—reflecting practical field procedures. Based on this framework, a dedicated evaluation dataset was constructed, and six representative portable forensic tools were empirically evaluated under a controlled Windows 10 (NTFS) environment simulating active system conditions. The experimental results demonstrate notable differences in tool capabilities across investigative phases. In the search phase, variations were observed in NTFS parsing and Windows artifact analysis, while the selection phase revealed disparities in file filtering, keyword search, encrypted file handling, and preview functions. In the seizure phase, only a subset of tools sufficiently supported evidence collection, integrity verification, and reporting requirements necessary for selective seizure. These findings highlight that no single tool uniformly satisfies all functional requirements, underscoring the need for context-dependent tool selection. The proposed framework and evaluation results provide practical guidance for digital forensic practitioners in selecting appropriate tools for selective seizure in field investigations. Moreover, this study contributes a reproducible methodological foundation for future research on selective seizure evaluation, supporting the development of more precise, proportionate, and legally robust digital evidence collection practices in Windows-based forensic investigations.

Keywords: digital evidence; digital forensic investigation; evaluation model; NTFS parsing; selective seizure functions; windows forensic tools; windows log analysis

1. Introduction

In digital forensics, search and seizure procedures typically involve the copying or printing of digital data, except physical storage media [1]. Despite the importance of selective seizures in investigations, research on the tools used during the process is lacking. Investigative agencies use forensic tools to search and seize storage media, and their legal admissibility depends on the ability of such tools to reliably perform selective seizures. However, digital forensic tools with selective seizure functions are yet to be developed, resulting in the insufficient implementation of relevant functionalities. An effective selective seizure function should offer the capability to extract and seize only digital information relevant to criminal suspicion, thereby minimizing unnecessary information,

 doi:10.20944/preprints202601.1696.v1

protecting privacy and human rights, ensuring compliance with laws, and enabling investigations that meet the required standards. This study identifies selective seizure functions, six digital forensic tools are selected, and a model to evaluate the functions is designed. Based on the model, a dataset is generated to test the capabilities of each tool. The evaluation criteria are designed based on the requirements for digital-forensic-tool verification from prior research, such as "NIST CFTT verification items and domestic TTA standardization documents." The experimental environment is deployed in Windows 10, which features a high domestic market share of 77.9% [2], and tools registered with the National Institute of Standards and Technology (NIST) are employed to support the portable, graphical user interface (GUI)-based system for the New Technology File System (NTFS) [3]. Specifically, we aim to evaluate the efficiency and practicality of digital forensic tools by simulating their actual execution in the field. To evaluate the selected tools, a dataset is developed for tests over three phases—search, select, and seizure—used to design the evaluation model. In the search phase, the file system information and Windows-log analysis related to file timestamps are considered to mirror actual field conditions, whereas messenger, cloud, remote program, and virtual environment analyses are excluded.

In the selection phase, tools are selected based on their ability to filter and search necessary files, excluding unallocated areas and slack space. The seizure phase focuses on selecting logical images using file-level collection functions, additional physical images, and memory-acquisition functions. The experiment is designed to consider the environment and systems employed for actual selective seizures in the field. The experimental setup for evaluating the functions of Windows forensic tools employs two laptops running Windows 10 in a VMware Workstation Pro virtual environment to simulate active system conditions [4]. The selected digital forensic tools are run in a portable format to evaluate their performance.

The remainder of this paper is organized as follows: Section II analyzes related work from both domestic and international sources on the concept and characteristics of digital evidence, legislative developments in criminal procedure law, and search and seizure procedures, including selective seizures. Additionally, the challenges and limitations associated with selective seizures of digital evidence are discussed. Section III presents the design of the proposed evaluation model for investigating selective seizure tools. A dataset is developed for evaluation and the experimental limitations are discussed. Section IV describes the experiment conducted using the designed evaluation dataset, and verifies whether the six tools support the analytical functions required in the evaluation model. Accordingly, recommendations on the use of tools depending on the type of crime are provided. Finally, Section V discusses the possible applications of the developed dataset and model, concludes the study, and presents future research directions.

Furthermore, with the increasing adoption of cloud computing and artificial intelligence technologies, digital forensics is evolving to address new challenges. Cloud forensics presents unique challenges in terms of data collection and analysis due to the distributed nature of cloud storage and the dynamic allocation of resources.[5,6] Recent advancements in AI-powered forensic tools have shown promise in automating the analysis of large-scale cloud environments,[7] enabling more efficient detection of relevant evidence across distributed systems. These tools can assist investigators in identifying patterns and relationships in cloud-stored data that might be difficult to discover through traditional methods. However, the integration of AI and cloud forensics also raises new considerations regarding data privacy, jurisdiction, and the validation of AI-assisted findings.

2. Related Work

Yoon and Lee [8] developed a system for the automated collection of crime-specific data from crime scenes and organized them by crime type. They presented an objective and standardized evidence-collection method and analysis guidelines independent of investigators' skills. They further developed an initial response method for field use. Cho[9] found that in a Windows file system, deleting a directory also deleted the files within it. When a file is deleted, the timestamps for writing, master file table (MFT) modification, and access time in the $Standard Information attribute change

 doi:10.20944/preprints202601.1696.v1

simultaneously, whereas the three creation times remain unchanged. Shin[10] examined the email search and seizure process employed at a crime scene and developed an effective tool for the detailed analysis of email attachments based on field conditions. Methods to efficiently conduct searches and seizures have been proposed, suggesting the requirement for field tools tailored to file characteristics, rather than focusing solely on the importance of post-search acquisition. Lee and Shim[11] reviewed general verification procedures for digital forensic tools and conducted performance evaluations using proposed verification items and functional requirements. Their findings revealed fragmented file recovery, file-type recognition, and the handling of Korean strings as areas requiring improvements. Hama and James[12] compared the characteristics, development release cycles, and development patterns of digital forensic imaging tools, and proposed methods to address maintenance vulnerabilities. Park et al.[13] proposed verification scenarios and datasets for partition-recovery tools and established the dependency of tool performance on their ability to recover data from damaged storage media. Quick and Choo[14] emphasized the storage of a plethora of personal information on digital devices. Based on advancements in digital forensic technologies, they proposed standard procedures aligned with legal frameworks to resolve conflicts between investigations and safeguard human rights. Digital forensic imaging techniques were applied to big data and an approach for selecting key files and data such as registries, emails, documents, spreadsheets, Internet history, communications, log files, photos, and videos for imaging was developed. The data volume of the original media was successfully reduced by 100 times. Furthermore, the method was applied to cases from the Australian Law Enforcement Agency, and the possibility of further data volume reduction was established, demonstrating the applicability of data reduction techniques for the selective seizure of digital forensic data. Kim et al.[15] proposed a dataset for verifying Windows forensic tools by classifying Windows artifacts into two categories: time-related and behavior-based artifacts, as test requirements for digital forensic tool verification. Lee[16] designed a software quality evaluation model based on the ISO/IEC 9126 standard and evaluated the reliability, usability, efficiency, maintainability, and portability of digital forensic tools.

Internationally, the U.S. NIST, a federal agency in the Department of Commerce, tests the functionality, reliability, and consistency of digital forensic tools to provide tool verification information to law enforcement agencies, investigative bodies, and corporate security professionals. Particularly, the CFReDS project provides datasets for testing digital forensic tools to enhance digital evidence analysis capabilities.[17] Additionally, the Digital Forensic Research Workshop (DFRWS) has researched digital forensic tools and techniques to conduct standardization studies.[18] Contrarily, Forensic Focus reviews forensic tools, conducts webinars, and hosts forums, thereby providing information related to cybercrime investigations. However, previous studies have either been conducted on Windows 10 or older versions or have not experimentally evaluated and verified digital forensic tools for selective seizures that can be deployed in the latest operating systems. Therefore, this experimental study is designed to reflect tool verification requirements according to recent changes in the technological environment. The selected evaluation items are categorized into individual test cases and tested in an environment that simulates selective seizure procedures.

3. Evaluation Model and Data Set Development

3.1. Evaluation Model Disign

Various institutions worldwide produce numerous forensic images annually to compare and verify the proficiency and performance of various forensic tools.[19,20] However, the existing image analysis method has certain limitations; although the method is suitable for analyzing seized media, it is inadequate for selective seizure analysis. Moreover, the method is static, whereas actual seizures in the field require dynamic image analysis because the operating system actively runs and imposes various restrictions such as the need for file-access permissions. Domestic forensic societies, universities, and other institutions have developed and used various datasets for digital forensic tool verification and expert qualifications.[21–23] However, to test tools for domestic environments, an

 doi:10.20944/preprints202601.1696.v1

analytical method that reflects the requirements of field investigations, rather than relying solely on media seizures, is required.[24–26] Additionally, an environment that allows the testing of additional tools in the future is desirable.[27,28] Instead of using an evaluation method based on image files, a dataset that can be tested in an actual field environment is crucial.[29,30] Furthermore, an evaluation model to test the functions of forensic tools is essential for ensuring technical accuracy and legal validity.[31,32]] The tests should be conducted in accordance with the standards established by government agencies, academic research institutions, and international standardization organizations.[33–35] Table 1 presents image formats used in the test datasets by institutions and prior studies.

Table 1. Digital-forensic-tool Datasets used in Previous Projects and Studies.

No.	Category	Project/Research	Format
1	Institution	Computer Forensic Reference Data Set (CFReDS) Project	E01, DD
2		Computer Forensic Tool Testing Project	E01, DD
3		Digital-forensic-tool Testing (DFTT)	DD
4		Naval Postgraduate School Digital Forensics Corpus (NPS Corpora)	File Dump, E01, DD, packet dump File
5		ForensicKB	E01, DD, File
6	Prior Research	ISO/IEC 9126-based Quality Evaluation Model for Digital Forensic Tools	Live
7		Reliability Evaluation through Verification of the Analytical Functions of Computer Forensic Tools	DD
8		Development of a Verification Dataset for Windows Forensic Tools	VMDK
9		Configuration of a Computer Forensic Tool Verification Image Suitable for Domestic Environments	
10		Reliability Verification of Evidence Analysis Tools for Digital Forensics	E01, DD, File

Existing requirements for verification covered in prior research were collected and compiled by borrowing the requirements required for the digital forensic tool evaluation model, and the digital forensic requirements are as follows: 1) file system verification requirements 2) file search verification requirements 3) file time related source log analysis verification requirements 4) application analysis verification requirements 5) data deletion and hidden verification requirements 6) verification and report requirements were used as evaluation indicators for verifying digital forensic analysis tools. [Table 2, 3, 4, 5, 6]

Table 2. File system validation requirements.

numbers	Verification requirements
1	Recognizes the specified file system and analyzes the metadata.
2	Recognizes and analyzes all files registered in the file system.
3	Analyze accurate time information for all files and folders.
4	Recognizes and processes fragmented files.

 doi:10.20944/preprints202601.1696.v1

5	Recognizes and analyzes all files and folders registered in the GUID partition.
6	Handles and recognizes incorrect partition information.
7	Shows the location on the digital source for all files.
8	Detects hidden partitions and recognizes all files and folders
9	Analytical tools provide data extraction and analysis functions for evidence media to identify evidence.
10	Analysis tool functions always output the same results when the same input data is given in the same environment.
11	Analysis tool functions keep input data revisions to a minimum and report modifications when they occur.
12	The analysis tool supports at least one file system and accurately recognizes the file systems and metadata supported by the tool.
13	The analysis tool records a log of events performed.
14	Analysis tool functions report errors that occurred during execution.

Table 3. File search validation on requirements.

Numbers	Verification requirements
1	The results of considering to the query are the same as the three matching the query.
2	Navigation is possible with one or more character encodings.
3	You can explore strings in Slack space.
4	If you specify a specific area within the evidence disk as the search range, results are output only at that point.
5	You can use regular expressions to navigate.
6	You can search with Hash Set (Hash Set).
7	You can explore the full spectrum of digital sources. (including slack, parity, deleted areas, and between files)
8	You can extract any file from a supported file system.
9	If a reorganization related error occurs while extracting a file, an error report is possible.
10	The collected files are collected to the original.
11	When extracting a file, unreadable parts are potentially as extractable data.
12	Users are considering when storage space is low.
13	Korean language support is available.
14	Invented the file system information contained in the validation data.
15	Preserves hash values (minimum MD5, SHA 1) for all files included in validation data.

 doi:10.20944/preprints202601.1696.v1

16	Combating the fragmentation status and count of some of the files included in the validation data.
17	Validates keyword searches in the NTFS file system.
18	Validates the ability to extract data from NTFS deleted files.
19	Verify the NTFS automatic detection feature.
20	Verify the data extraction function in the basic data carving function.
21	If a subset of the string search function is specified, results are output only from that subset.
22	A limit on the number of matches is applied when searching for sorted/unsorted subsets in a string search.
23	The screen is displayed according to the specified text direction.
24	Synonym search is supported.
25	When performing a fuzzy search, it must match a close misspelling (close misspelling) in the query string.
26	When performing a phonetic search (phonetic search), it must match words pronounced the same as the query string.
27	If you provide a predefined query, the response returned must be the same set of matches for the query.
28	It must support logical operations such as And, or, and not.
29	Analytical tools should provide data extraction and analysis capabilities to identify evidence.
30	You must be able to browse regardless of capitalization.

Table 4. Application analysis verfication requirements.

numbers	**Verification content**
1	Log file analysis must be able to recognize at least one or more log file formats and provide a function to search and filter desired events.
2	It is necessary to be able to analyze time information on internal data according to the environment in which the original was created.
3	System/user settings information analysis must be able to analyze information about the operating system, information about users, and information about installed applications.
4	Log file analysis must be able to accurately recognize and report on various log file formats through documentation provided by the tool (including usage, purpose, operating mechanism, and system requirements).

Table 5. Data deletion, concealment verification requirements.

numbers	Verification content
1	The deleted file recovery function must support the recovery function in a file system confirmed by documents provided by the tool (a set of materials describing usage, purpose, operation, system requirements, etc.).

 doi:10.20944/preprints202601.1696.v1

2	The deleted file recovery function must identify all deleted file system objects that can be recovered from metadata maintained after the file system object has been deleted.
3	The deleted file recovery function must report errors that occurred in constructing recovered objects.
4	The deleted file recovery function must configure a recovered object for each deleted file system object from the remaining metadata.
5	Each recovered object must include all unallocated data blocks identified in the remaining metadata.
6	Each recovered object must consist only of data blocks from the deleted block pool.
7	If the deleted file recovery function generates estimated content, the recovered object must be composed of data blocks from the original file system object identified in the remaining metadata.
8	If the deleted file recovery function generates estimated content, any data blocks in the recovered object must be organized in the same logical order as the original file system object identified in the remaining metadata.
9	If the deleted file recovery function generates estimated content, the recovered object must consist of the same number of blocks as the original file system object.
10	Verify the FAT deleted file extraction function.
11	Verify the NTFS deleted file extraction function.
12	Verify the basic data carving function.

Table 6. Verification/Report Requirements.

numbers	Verification content
1	The analysis tool must output the results of the analysis performed in the form of a report.
2	The hash values of all files belonging to the verification data are preserved. At this time, the hash function uses at least MD5 and SHA1.
3	Analysis tool functions must always output the same results when given the same input data in the same environment.
4	Timeline analysis must be able to extract information such as creation, modification, access time, and ownership of a file through MAC (modified, modified, change of status) time analysis, and must be able to sort and list in chronological order using this information.
5	Analysis tool functions must report errors that occurred during execution.

This verification design method is the result of a digital forensic field response tool evaluation model design study not covered in prior research.

Previous studies have established evaluation content for digital forensic tools, and the requirements for selective seizure tools have been reflected in their designs. The requirements include:

Tools must be portable, field-ready, feature a GUI, and easy to use.

Tools must include functions for collecting case-relevant files in the field.

Tools must be able to analyze various logs.

Tools should support result verification and reporting.

During selective seizures, the Windows file system should be searched and the generated evidence must be verified. Once the search is complete, the Windows log information should be

 doi:10.20944/preprints202601.1696.v1

searched for files of interest, and those related to criminal allegations have to be identified. Finally, relevant files must be collected, and a report should be generated. Accordingly, an evaluation model for the Windows forensic selective seizure tool was designed for this study, as shown in Figure 1.

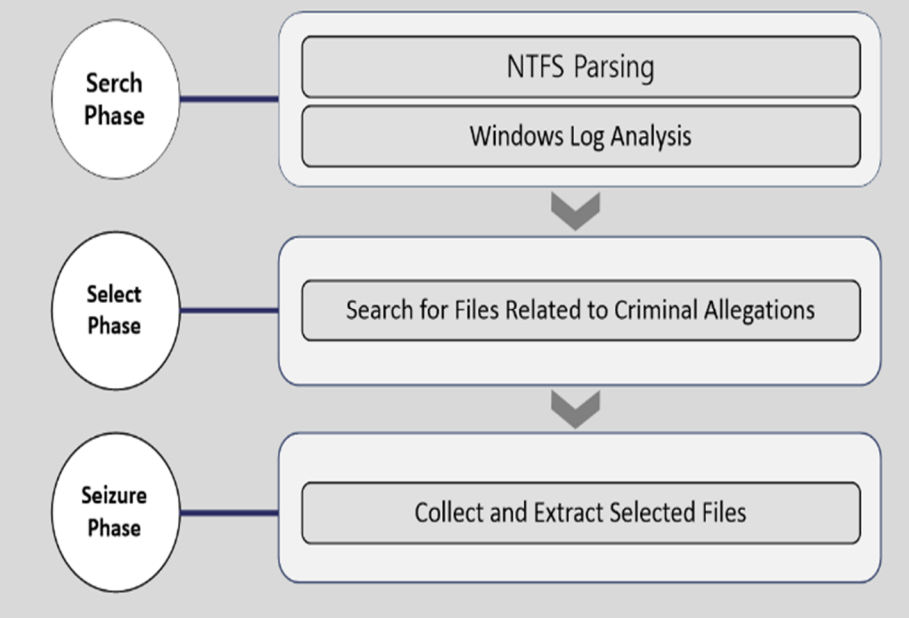

Figure 1. Design process of the selective-seizure-tool evaluation model.

The digital forensic process involves the following steps: 1. investigation preparation; 2. evidence acquisition, 3. transportation and storage, 4. investigation and analysis; and 5. report generation. In this study, a new evaluation model was developed for window-based selective seizure tools that consider field-response criteria. To ensure comprehensive testing, detailed evidence image files representing diverse application environments were created. File integrity was verified to ensure that the content did not change after the analysis, considering the characteristics of the target tools. The evaluation involved comparing the analysis functions of the selected seizure tools across the three main phases and eight items, as listed in Table 7.

Table 7. Phases and Items Employed for Evaluating the Analysis Functions of Selective Seizure Tools.

No.	Phase	Evaluation Item
1	Search	• NTFS Parsing (BitLocker analysis, Partition MFT, VBR (MBR, GPT) deletion, damage analysis, $MFT, $MFT Entry, $Standard_Information, $FileName, $MFT deletion information) • File- and Folder-access Log Analysis (JumpList, Link File, Shellbag MRU, Windows.edb, ActivitiesCache.db, etc.) • Application-execution Log Analysis (Prefetch, ActivitiesCache.db, AmCache, SRUMDB) • Data-transmission Log Analysis (Setupapi.log, Registry, EventLog, AmCache) • Information Search Log Analysis (Chrome, Edge, Whale: History, Download, Password, Cache, Cookies) • Suspicious-activity Log Analysis (EventLog: System ON/OFF, User Account Changes, System Time Changes)

 doi:10.20944/preprints202601.1696.v1

2	Select	• Information Search (File and folder-name search, file keyword analysis, file-header classification, DRM files, encrypted files, etc.)
3	Seizure	• Collection and Extraction (Logical image, physical image, memory acquisition, report generation, hash verification, error handling report)

3.2. Evaluation Model Disign

Various institutions worldwide produce numerous forensic images annually to compare and verify the proficiency and performance of various forensic tools.[19,20] However, the existing image analysis method has certain limitations; although the method is suitable for analyzing seized media, it is inadequate for selective seizure analysis. Selective seizure in digital forensic investigations requires not only functional support from forensic tools but also a systematic method for evaluating whether such tools adequately satisfy procedural, technical, and legal requirements. To address this need, this study formalizes the evaluation of selective seizure functions through a phase-based analytical model supported by quantitative expressions. The proposed evaluation model is structured according to the three procedural phases commonly observed in field investigations: search, selection, and seizure. Each phase reflects a distinct investigative objective—identifying relevant artifacts, narrowing the scope of evidence, and securely collecting admissible digital evidence. Rather than treating selective seizure as a monolithic process, this phased structure allows the functional capabilities of forensic tools to be examined in alignment with actual investigative workflows , provides a narrative explanation of the rules applied for calculating the Selective Seizure Capability Index (SSI) and presents a step-by-step expansion of the calculation results for each evaluated Windows forensic tool. First, the SSI calculation begins with symbol-based scoring. In the evaluation tables, each functional item is assessed using qualitative symbols. These symbols are systematically converted into numerical scores to enable quantitative analysis. Specifically, a symbol of O is assigned a score of 3, □ is assigned a score of 2, △ is assigned a score of 1, and – is assigned a score of 0. Accordingly, for each evaluation item i within an investigative phase p, the score of a tool T is defined as $s_{p,i}(T) \in \{3, 2, 1, 0\}$. Second, phase-level raw scores are calculated. For each investigative phase—Search, Select, and Seizure—the raw score $E_p(T)$ of a tool is obtained by summing the scores of all evaluation items belonging to that phase. Because the number of evaluation items differs across phases, raw scores alone cannot be compared directly. Third, to ensure fairness across phases, each phase-level raw score is normalized. The normalization is performed by dividing the raw score by the maximum possible score for that phase, which is defined as three times the number of evaluation items. As a result, the normalized phase score ranges between 0 and 1, representing the proportion of functional requirements satisfied by the tool within that phase. Fourth, the normalized phase scores are aggregated into a single SSI value through weighted summation. In this study, weights of 0.3 were assigned to the Search and Select phases, while a higher weight of 0.4 was assigned to the Seizure phase to reflect its direct relevance to evidentiary integrity and admissibility. The resulting SSI value therefore represents the overall degree to which a tool supports selective seizure procedures.

For interpretation convenience, the final SSI values are also expressed as percentages by multiplying the normalized SSI by 100. In this study, the number of evaluation items and corresponding maximum scores were fixed as follows: the Search phase consisted of 49 items with a maximum score of 147, the Select phase consisted of 80 items with a maximum score of 240, and the Seizure phase consisted of 10 symbol-based items with a maximum score of 30. Non-symbolic entries, such as time measurements or descriptive report formats, were excluded from the SSI calculation to maintain reproducibility. Based on these rules, Tool A achieved a low SSI value, primarily due to minimal support in the Search and Select phases, despite moderate performance in the Seizure phase. Tool B demonstrated balanced performance across all phases, resulting in a moderate SSI. Tool C showed

 doi:10.20944/preprints202601.1696.v1

relatively strong performance in the Select and Seizure phases, which significantly contributed to its higher SSI. Tool D exhibited strong Search and Seizure capabilities but weaker Select-phase support, leading to a mid-range SSI value. Tool E achieved the highest SSI, reflecting consistently strong performance across all three phases, particularly in Search and Seizure functions. In contrast, Tool F showed limited Search and Select capabilities and relied mainly on Seizure-phase performance, resulting in a comparatively low SSI. Overall, this narrative calculation process demonstrates that the SSI is not a simple ranking score but a structured and transparent index derived from phase-specific functional performance. By integrating normalization and weighted aggregation, the SSI enables reproducible, fair, and procedurally aligned evaluation of Windows forensic tools in selective seizure scenarios. The proposed evaluation model is structured according to the three procedural phases commonly observed in field investigations: search, selection, and seizure. Each phase reflects a distinct investigative objective—identifying relevant artifacts, narrowing the scope of evidence, and securely collecting admissible digital evidence. Rather than treating selective seizure as a monolithic process, this phased structure allows the functional capabilities of forensic tools to be examined in alignment with actual investigative workflows. Let T denote a Windows-based forensic tool subject to evaluation, and let p ∈ {Search, Select, Seizure} represent an investigative phase. The evaluation score of tool T at phase p is defined as:

$$E_{p(T)} = \sum_{\{i=1\}}^{\{n_p\}} s_{\{p,i\}(T)}$$

where n_p is the number of evaluation items associated with phase p, and s_{p,i}(T) denotes the score assigned to the i-th evaluation item. This formulation enables a structured aggregation of tool capabilities within each procedural stage, ensuring that individual functional elements are assessed systematically rather than anecdotally. To preserve the intuitive clarity of qualitative assessment while enabling quantitative comparison, this study maps the symbol-based evaluation scheme to a numerical scoring function. Specifically, the score s_{p,i}(T) is defined as follows:

s{p,i}(T) = 3, if the tool fully satisfies all verification criteria (O);

s{p,i}(T) = 2, if three or more criteria are satisfied (□);

s{p,i}(T) = 1, if criteria are partially satisfied with manual intervention (△);

s{p,i}(T) = 0, if the criteria are not satisfied (-).

This mapping allows qualitative judgments to be expressed in a reproducible and transparent manner, thereby reducing subjectivity and enhancing analytical rigor. At the same time, the original symbolic representation is retained in result tables to maintain practical interpretability for forensic practitioners. While phase-specific evaluation scores provide detailed insights into tool performance, selective seizure in practice requires an integrated assessment that reflects procedural priorities. Accordingly, this study introduces the Selective Seizure Capability Index (SSI) as a composite measure of overall tool suitability:

$$SSI(T) = \sum_{\{p \backslash in\ P\}} w_p \cdot E\_p(T)$$

where P = {Search, Select, Seizure} and w_p denotes the weight assigned to phase p, subject to the constraint that ∑_{p} w_p = 1. The weighting scheme allows the evaluation model to adapt to investigative contexts; for example, greater emphasis may be placed on the seizure phase due to its direct implications for evidentiary integrity and admissibility, while search and selection phases emphasize efficiency and proportionality. By integrating phase-based evaluation, symbol-to-score formalization, and weighted aggregation, the proposed model establishes a coherent and extensible framework for assessing selective seizure capabilities in Windows forensic tools. This formalization not only enhances the reproducibility of comparative evaluations but also provides a methodological foundation for future extensions, such as usability metrics, automation levels, or cross-platform adaptations. Moreover, the method is static, whereas actual seizures in the field require dynamic image analysis because the operating system actively runs and imposes various restrictions such as the need for file-access permissions. Domestic forensic societies, universities, and other institutions

 doi:10.20944/preprints202601.1696.v1

have developed and used various datasets for digital forensic tool verification and expert qualifications.[21–23] However, to test tools for domestic environments, an analytical method that reflects the requirements of field investigations, rather than relying solely on media seizures, is required.[24–26] Additionally, an environment that allows the testing of additional tools in the future is desirable.[27,28] Instead of using an evaluation method based on image files, a dataset that can be tested in an actual field environment is crucial.[29,30] Furthermore, an evaluation model to test the functions of forensic tools is essential for ensuring technical accuracy and legal validity.[31,32]] The tests should be conducted in accordance with the standards established by government agencies, academic research institutions, and international standardization organizations.[33–35] Table 8 presents Tools evaluated in this study.

Table 8. Portable Tools Evaluated in this Study.

No.	Category	Tool Name	Release and First Version
1	Commercial	A	2019
2	Non-commercial	B	2010
3	Commercial	C	1995
4	Commercial	D	2011
5	Commercial	E	2011
6	Commercial	F	2021

To maintain objectivity and avoid bias toward specific products, the tool names were anonymized[44,45], and their version information was not included to ensure anonymity. However, the latest version, as of November 30, 2023, was used. Anonymization ensured that the focus remained on the evaluation results and comparative analysis of tool functions rather than on a particular product. The evaluation results were expressed using four symbols, as listed in Table 9, with the recommended tools labeled Recommendation 1 (Tool Name*)" and Recommendation 2 (Tool Name**).

Table 9. Symbols for Presenting the Evaluation Results of Selective Seizure Tool Functions.

Symbol	Description
O	Meets all details of the verification items
□	Meets three or more details of the verification items
△	Meets three or more details of the verification items but requires manual analysis
-	Does not meet any detail of the verification items

3.3. Dataset Development

To simulate a typical investigative scenario, the experimental environment for evaluating selective seizure functions comprised two standard laptops (Intel i5-1035G4 CPU, 8 GB RAM, and 250 GB SSD) instead of high-performance systems. The environment included a laptop with Windows 10 64-bit installed on a VMware virtual machine. Detailed specifications are listed in Table 10.

Table 10. Laptop Specifications and VMware Virtual Machine Environment Used for the Experiments.

No.	Type		Specifications
1	Laptop	CPU	Intel Core i5-1035G4 CPU @ 1.10 GHz 1.50 GHz
		RAM/HDD	8 GB/Mtros SSD NVMe M.2
2	Virtual Machine	Windows	Windows 10 PRO 64-bit 22H2 NTFS configuration

 doi:10.20944/preprints202601.1696.v1

	Version	VMware Workstation PRO 17.5
	CPU/RAM	4 Core/4 GB

The method for evaluating selective seizure function comprised three phases: search, selection, and seizures. To minimize the number of steps required, the recovery phase was integrated into the file system and application analysis. Additionally, based on the previously described procedures, the evaluation components included NTFS analysis, Windows-log analysis, information searches, and collection/extraction. The parameters tested in the VMware virtual machine are listed in Table 11.

Table 11. Details of the Evaluation Items.

Phase	Parameter	Description
Search	Portable	Verify the execution and automated analysis functionality of portable GUI-based tool selection
	BitLocker identification and support for decryption after entering the password	Verify support for recognizing BitLocker-encrypted partitions on Windows and unlocking and decrypting them after entering the recovery key or password
	Partition Damage	Verify support for recovering deleted or damaged partitions master boot record (MBR), GUID partition table (GPT) (MFT, volume boot record (VBR))
	$MFT Analysis	Verify support for recognizing NTFS and parsing $MFT. ($MFT structure parsing, size information)
	$MFT attribute timestamp analysis ($SI, $FN)	Verify recognition of $MFT and support for file-information and metadata analysis. ($SI creation, modification, access, MFT modification, attribute flag information, $FI parent directory file reference address, creation, modification, access, MFT modification, file allocated size, actual file size, attribute flag, name length, name type, name information support)
	$MFT Deletion	Verify deleted record information in $MFT entries. (Unallocated file, file name, creation, modification, access)
	JumpList	Verify document file-access records. (Analysis count, file name, link creation, link modification, link access, target creation, target modification, target access, volume name, volume S/N, size, path, original path)
	Link File	Verify document file-access records. (Analysis count, file name, link creation, link modification, link access, target creation, target modification, target access, volume name, volume S/N, size, path, original path)
	Shellbag MRU	Verify folder-access records. (Analysis count, folder name, visit time, creation time, access time, modification time, path, original path)

 doi:10.20944/preprints202601.1696.v1

	ActivitesCache.db	Verify document file-access records. (Analysis count, display text, last modification time, app ID, path, original path)
	Windows.edb	Verify document file records. (Document count, file name, creation, modification, file type, file path, content, original path)
	Volume Shadow	Verify backup records of three documents. (Analysis count, file name, creation time, original path)
	$Logfile	Verify document file records (Analysis count, events, original path)
	$UsnJrul	Verify document file records (Analysis count, events, original path)
	Thumbnail.db	Verify thumbnail cache image file records. (Image name, original path)
	$Recycle.bin	Verify records of deleted files (File count, $R, $I)
	Prefetch	Verify records of executed programs. (Analysis count, program name, execution time, execution count, path, original path)
	Program Execution	Verify records of executed programs. (Analysis count, program name, execution time, execution count, user, path, original path)
	AmCache	Verify record of executed program. (Key time, key name, name, publisher, size, original path)
	SRUMDB	Verify resource-usage records of executed programs. (Creation time, program, original source)
	External Storage Device	Registry: Verify external storage device records. (Model name, first connection time, last connection time, disconnection time, serial number, volume name, original path)
		Setupapi.log: Verify external storage device records. (Model name, first connection time, original path)
		EventLog: Verify external storage device records. (Event ID, connection/disconnection time, device information, original path)
	Network Connection	Wireless: Verify wireless network records. (Network name, last connection time, original path)
		Wired: Verify wired-network records. (Adapter name, IP, Subnet mask, DHCP,

 doi:10.20944/preprints202601.1696.v1

			original path)
		Web Access	Verify internet-access records. (Analysis count, accessed website, access time, original path)
		Search Terms	Verify web search term records. (Analysis count, search terms, access time, original path)
	Web Browser (Chrome, Edge, Whale)	Downloads	Verify records of files downloaded from the internet. (Analysis count, website, downloaded file, download time, original path)
		Passwords	Verify saved web ID/password records. (Save count, creation time, website, original path)
		Cache	Verify web-cache file records. (Analysis count, cache file, access time, website, original path)
		Cookies	Verify internet cookie records. (Analysis count, cookie file, access time, website, original path)
		System ON/OFF	Verify system ON/OFF records. (ON time, OFF time, computer name, EventID, original path)
	EventLog	User Account Changes	Verify system user account change information. (Account name, event time, EventID, original path)
		System Time Changes	Verify system time change information. (Previous time, new time, user ID, EventID, original path)
Select	File-name Search		Verify search of Korean file names. “Network Diagram.pptx, Risk Burden.pdf, Office Lease Contract.hwp” Verify support for logical operators AND, OR. Verify support for regular-expression searches. “Forensic[, -]Science, Sungkyunkwan[0-9가-힣]University, http://www\.[가-힣]+\.com, C:\\Images\\KakaoTalk\\.gif, 02[-)]*3290-1212”
	Keyword Analysis		Verify search for keywords “forensics, digital forensics, selection, tools, Sungkyunkwan University” in document file extensions. (cell, csv,doc,docx,hwp,hwpx,pdf,pptx,rtf,show,txt,xls,xlsx)
	File-header	Documents	cell, csv, doc, docx, hwp, hwpx, pdf,

 doi:10.20944/preprints202601.1696.v1

	Identification		pptx, rtf, show, txt, xls, xlsx.
	Preview	Digital Rights Management (DRM)	Verify the identification of Fasoo, SoftCamp, and Markany.
		Encrypted Files	Verify the identification of doc, docx, hwp, ppt, pptx, 7z, tar, and zip.
		Documents	cell, csv, doc, docx, hwp, hwpx, pdf, pptx, rtf, show, txt, xls, xlsx.
		Compound Files	Verify the identification of pst, ost, sqlite, db, 7z, zip, tar, egg, and rar.
		Images	Verify the identification of awd, psd, dwg, bmp, png, psp, jpeg, and jpg.
Seizure	Logical Image	Verify support for logical image acquisition.	
	Physical Image	Verify support for physical image acquisition.	
	Memory Collection	Verify support for memory-dump collection.	
	Report	Verify the generation of a report for logical image acquisition (file name, path).	
	Hash Verification	Verify the inclusion of hash information in the report and support for calculating at least two hashes.	
	Error Handling Log	Support for logging errors occurring in the tool and other logs.	

The information on the image files created in the VMware test environment included name, size, and hash information. The test environment was configured to generate the data necessary for evaluating the tool performance, including partition damage, file deletion, BitLocker configuration, and file timestamp information from Windows logs. A logical image comprising 200,000 files was created to test the selective seizure performances of the tools. The details of the virtual machine disk (VMDK) dataset are presented in Table 12.

Table 12. VMDK Dataset.

Verification Item	File Name	File Size	File Hash Value (MD5, SHA1)
Search	MBR-000002.vmdk	17 GB	d666ad43460de6f9955e4e045b10ba8e 54608a7ea3437e3ea6952b5722cfe4baa52fa2da
	Basic Sample Machine-cl1.vmdk	22.9 GB	62ac1cfdea9a1d3b13395c6eb9c9ecc8 e3e33f24909b56a5fc57491b17621310d8a76a66
Search, Select	Windows 10 x64(1). vmdk	61.9 GB	d666ad43460de6f9955e4e045b10ba8e 54608a7ea3437e3ea6952b5722cfe4baa52fa2da
Seizure	Basic Sample Machine-cl1.vmdk	25.3 GB	b11f85b705402c86cce19ea3ee8dec0b f2925883e67c719f01ae57762d82cd8699a1d206

The test dataset used in the VMware environment was divided into three configurations: file-system and Windows-log analyses, file-search, and collection and extraction. Additionally, as numerous files and substantial Windows log information were used, along with processes such as installation, creation, and deletion, four test environments were constructed. Each of the six tools was evaluated once, and the VMware snapshot functionality was used to ensure a consistent evaluation of the results. The experimental procedure, as illustrated in Figure 2, was implemented in the

 doi:10.20944/preprints202601.1696.v1

VMware environment on a laptop using a custom evaluation dataset that was saved as a VMDK image file for reuse.

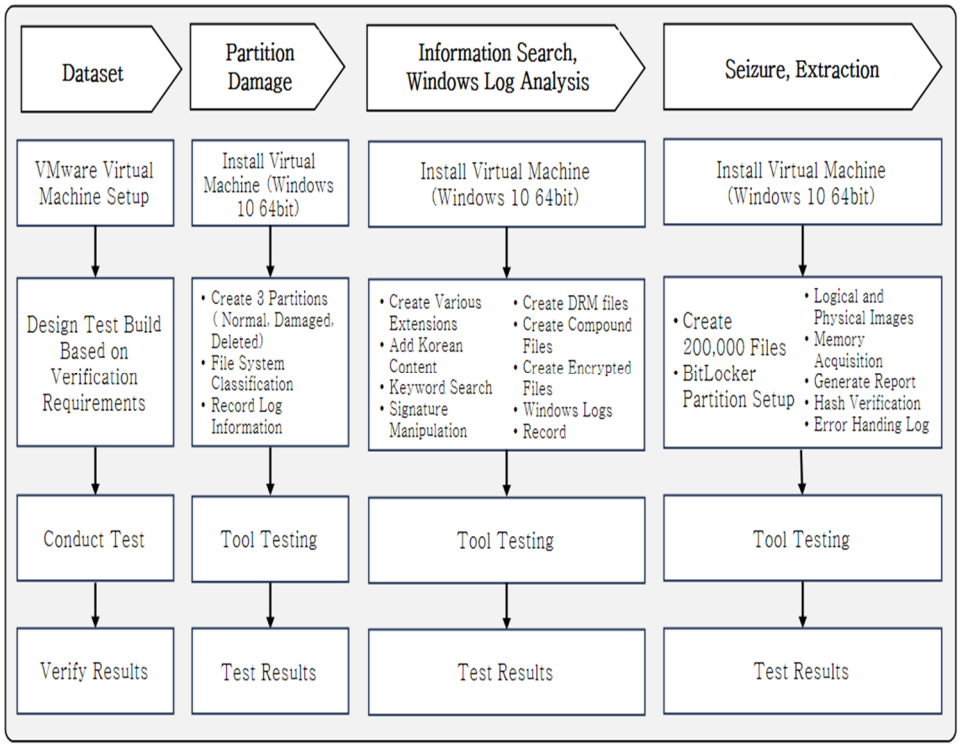

Figure 2. Experimental procedure for evaluating selectice seizure functions of the six tools.

The dataset employed to evaluate selective seizure functions enabled extensive testing of portable tools and yielded accurate results through four configurations of Windows functions. The dataset offered a key advantage of data from Koreans, thereby allowing tool verification for application to the domestic environment, supporting the Korean language.

4. Results and Discussion

4.1. Search Phase

4.1.1. NTFS Parsing

The NTFS parsing ability of the tools was tested across eight functions, as listed in Table 13. Tool C demonstrated the best performance, followed by Tool B. However, none of the tools met all requirements. For example, only Tool E recognized the BitLocker volume and supported key-input decryption, whereas Tool D supported the VBR (MBR) damage repair functionality. Additionally, Tools B and C supported the recovery of the deleted MFT (MBR) partitions. Tool B provided comprehensive support for analyzing $MFT entry headers, $Standard Information, $File Name, $Attribute, and $Date. Although Tool C met most evaluation requirements, Tool B proved valuable, as it provided substantial information concerning NTFS. The evaluation results listed in Table 13 indicated whether the tools accurately interpreted the file system and provided the necessary information.

Table 13. NTFS Parsing Results.

No.	Evaluation Item	Tool					
		A	B**	C*	D	E	F
1	BitLocker Support	-	-	-	O	-	-

 doi:10.20944/preprints202601.1696.v1

2	$MFT Analysis	O	O	O	O	O	O
3	$MFT Recognition	-	O	□	□	□	-
4	MBR (VBR) Damage	-	-	-	O	-	-
5	MBR (MFT) Partition Deletion	-	O	O	-	-	-
6	GPT (VBR) Damage	-	-	O	-	-	-
7	GPT (MFT) Partition Deletion	-	-	O	-	-	-
8	MFT File Deletion	-	O	O	-	O	-

*; **.

4.1.2. File-and Folder-Access Log Analysis

We evaluated the tools' ability to analyze file and folder-access logs across 10 evaluation items, as listed in Table 9. Tool E exhibited the best performance, followed by Tool D. In the link file analysis, Tool D provided comprehensive information, including the creation, modification, and access times for both the link and target files. However, Tool B provided some Korean file names as garbled characters, indicating unoptimized tool behavior for Korean and the requirement of a code page change function for proper display. Notably, only Tool E supported Windows.edb, $Logfile, and UsnJrnl, which contained critical information related to document and file activities, suggesting that the functionality of the other tools required improvements. Furthermore, analyzing folder-access logs was essential for confirming the evidence related to a case and tracing deleted information. The evaluation results, as listed in Table 14, indicated the importance of data activity analysis, which can be used to identify meaningful information and behaviors through prior investigations and evidence.

Table 14. File-and Folder-access Log Analysis Results.

No.	Evaluation Item	Tool					
		A	B	C	D**	E*	F
1	JumpList	-	-	△	□	□	-
2	Link File	□	□*	△	O	□	□
3	Shellbag MRU	-	□	△	□	O	-
4	ActivitesCache.db	-	□	-	O	O	-
5	Windows.edb	-	-	-	-	O	-
6	Volume Shadow	-	-	△	-	-	-
7	$Logfile	-	-	-	-	O	-
8	$UsnJrul	-	-	-	-	O	-
9	Thumbnail.db	-	-	-	O	O	-
10	$Recycle.bin	-	O	O	-	O	-

* Some issues in displaying Korean text; **.

4.1.3. Application-Execution Log Analysis

The tools' ability to analyze application-execution logs focused on four key areas, as listed in Table 10. Tool E performed the best, followed by Tool D. The results indicated that all tools supported the prefetch analysis, but Tool D did not support the SRUMDB analysis. Starting with Windows 10, log checking for application compatibility in the Cache and DB information were added. Although

 doi:10.20944/preprints202601.1696.v1

the logs provided valuable forensic insight, our evaluation revealed that some tools required improvements. Specifically, Tool B required improvements in Gram-execution log analysis, and Tool D required improvements in SRUMDB analysis. The evaluation results, as presented in Table 15, indicated the capabilities of the tools for analyzing application-execution activities, which could be used to determine the time and order of frequently used applications, detect data deletion or anti-forensic activities, and analyze application-execution behaviors.

Table 15. Application-execution Log Analysis Results.

No.	Evaluation Item	Tool					
		A	B	C	D**	E*	F
1	Prefetch	□	O	△	O	O	□
2	Program Execution	-	-	△	O	O	-
3	AmCache	-	△	△	O	O	-
4	SRUMDB	-	□	△	-	O	-

*; **.

4.1.4. Data-Transmission Log Analysis

The tools' ability to analyze data-transmission logs focused on six areas, as listed in Table 16. Tool C exhibited the best performance, followed by Tool E. However, Tool C necessitated manual analysis to verify the results, whereas Tool E enabled automatic analysis, offering greater usability. However, event logs related to external storage devices were not analyzed. Additionally, two tools supported the analysis of wired-network information, whereas two supported the analysis of the connection and disconnection records of external storage devices. In cases involving technology leaks, the history of the external storage device usage and network information required verification. Network information is categorized into wired and wireless, whereas external storage device information is analyzed using the registry, event logs, and Setup.API logs.

Table 16. Data-transmisson Log Analysis Results.

No.	Evaluation Item	Details	Tool					
			A	B	C*	D	E**	F
1	Network	Wireless	-	-	△	O	O	-
2	Connection	Wired	-	-	△	-	O	-
3		Registry	-	O	△	O	O	-
4	External Storage	Event Log	-	-	△	O	-	-
5	Device	Setup.API	-	-	△	△	O	-
6		AmCache	-	△	△	O	O	-

*; **.

4.1.5. Internet Search Log Analysis

The tools' ability to analyze Internet search logs was evaluated using 18 items across three browsers (Google Chrome, Microsoft Edge, and Naver Whale). The results are listed in Table 17, where Tool E performed the best, followed by Tool B. Additionally, only Tool E supported the Whale browser analysis, whereas both Tools D and E supported the password analysis of the Edge. As the information stored in browsers i saved in database (DB) format, the tools should support DB analysis. Additionally, cache and cookie information were not configured in dedicated modules, making the analysis results difficult to interpret; thus, functionality improvements in these areas were necessary.

 doi:10.20944/preprints202601.1696.v1

As web browsers store data on internet activities and the interests of users, they provide crucial information for criminal investigations.

Table 17. Internet Search Log Analysis.

No.	Browser	Evaluation Item	Tool					
			A	B	C**	D	E*	F
1		Web Access	-	O	△	O	O	-
2		Search Terms	-	O	△	O	O	-
3	Google	Downloads	-	O	△	O	O	-
4	Chrome	Passwords	-	-	-	-	-	-
5		Cache	-	O	△	-	O	-
6		Cookies	-	O	△	-	O	-
7		Web Access	-	O	-	O	O	-
8		Search Terms	-	O	-	O	O	-
9	Microsoft	Downloads	-	O	-	O	O	-
10	Edge	Passwords	-	-	-	O	O	-
11		Cache	-	O	-	-	O	-
12		Cookies	-	O	-	-	O	-
13		Web Access	-	-	-	-	O	-
14		Search Terms	-	-	-	-	O	-
15	Naver	Downloads	-	-	-	-	O	-
16	Whale	Passwords	-	-	-	-	-	-
17		Cache	-	-	-	-	O	-
18		Cookies	-	-	-	-	O	-

*; **.

4.1.6. Suspicious-Activity Log Analysis

The tools' capability to analyze suspicious activity logs was tested across three EventLog items, as listed in Table 18. Evidently, Tool C performed the best, followed by Tools D and E, whereas the other tools did not support EventLog analysis. The logs recorded a wide range of information and were the key to forensic analysis. Therefore, the functionality of the tools for analyzing EventLogs required improvements. Tool C allowed manual analysis by installing an additional viewer; however, searching for the results required knowledge of the real log ID. Therefore, the ability to interpret EventLog analysis more easily required enhancement. Error and system-change information recorded in EventLogs were used for suspicious-activity analysis and could be vital for identifying specific issues or proving criminal allegations.

Table 18. Suspicious Activity Analysis Results(Eventlog).

No.	Evaluation Item	Tool					
		A	B	C*	D**	E**	F
1	System ON/OFF	-	-	△	O	O	-
2	User Account Changes	-	-	△	-	-	-

 doi:10.20944/preprints202601.1696.v1

3	System Time Changes	-	-	△	-	-	-

*; **.

4.2. Search Phase

4.2.1. File-and Folder-Name Search

The tools' capability to analyze file and folder-name searches involved three items, as listed in Table 19. Tool C performed the best, followed by Tool E. Additionally, although most tools supported file name searches, they did not support folder name searches. Therefore, folder names and regular-expression search functionalities required improvements. In the NTFS, information on all files and directories, including records and file names, is stored in the MFT. The MFT record contains the $FILE_NAME attribute where the file name is stored.

Table 19. File and Folder-name Search Analysis Results.

No.	Evaluation Item	Tool					
		A	B	C*	D	E**	F
1	File Name	-	O	O	O	O	-
2	Folder Name	-	-	O	-	-	-
3	Regular Expression	-	-	-	-	□	-

*; **.

4.2.2. File-and Folder-Name Search

The evaluation of the tools' capability to support keyword search involved 14 file extensions and five Korean word items, as listed in Table 15. Tool E performed the best, followed by Tool C. Additionally, Tool D supported the index search but not the keyword search, whereas Tools A and B detected only one and two keywords, respectively. Additionally, the functionality of the tools for analyzing Korean keywords and supporting a wider range of file extensions required improvements. A keyword search was crucial for efficiently finding important information within large datasets, accelerating the investigation, and quickly accessing evidence critical for solving the case.

Table 20. File and Folder-name Search Analysis Results.

No.	File Type	Tool					
		A	B	C**	D	E*	F
1	cell	-	□	O	-	O	-
2	csv	-	O	O	-	O	-
3	doc	- 1 hit	- 2 hits	O	-	O	-
4	docx	- 1 hit	- 2 hits	O	-	O	-
5	hwp	-	O	O	-	O	-
6	hwpx	-	-	-	-	O	-
7	pdf	-	O	O	-	O	-
8	ppt	- 1 hit	O	O	-	O	-

 doi:10.20944/preprints202601.1696.v1

9	pptx		□	O	-	O	-
10	rtf	- 1 hit	- 2 hits	-	-	O	-
11	show	-	□	-	-	O	-
12	txt	-	□	O	-	O	-
13	xls	-	O	O	-	O	-
14	xlsx	-	O	O	-	O	-

*; **.

4.2.3. File-Header Classification

The file header classification performances of the tools were tested across 14 file extensions (Table 21). Tools B and C performed the best, followed by Tool E. However, none of the tools could classify the header information of all files, and Tools B and C did not support the same extensions. Moreover, only one tool supported the csv and hwpx extensions. The experiment involved altering the primary document extensions used in South Korea. The results of the file-signature analysis revealed some false positives, wherein files with the same signature were classified as the same document type. Classifying files based solely on file-header signatures led to a higher false-positive rate; therefore, future improvements should include comparing byte sequences and format structures unique to each file to accurately determine its original extension. Because file extensions are simply names indicating the information they contain, proper signature interpretation was necessary for accurate file recognition.

Table 21. File-header Classification Results.

No.	File Type	Tool					
		A	B*	C*	D	E**	F
1	cell	-	-	-	-	O	-
2	csv	-	-	O	-	-	-
3	doc	-	O	O	-	-	-
4	docx	-	O	O	-	O	-
5	hwp	-	O	O	-	O	-
6	hwpx	-	-	-	-	O	-
7	pdf	-	O	O	O	O	-
8	ppt	-	O	O		-	-
9	pptx	-	O	O		-	-
10	rtf	-	O	O	-	0	-
11	show	-	-	-	-	-	-
12	txt	-	O	-	O	-	-
13	xls	-	O	O	-	O	-
14	xlsx	-	O	O	-	-	-

*; **.

4.2.4. File-Header Classification

The DRM file identification performance of the tools was tested across ten items, as listed in Table 22. Evidently, Tool E performed the best; however, the tool identified only five items, whereas the others could not identify one item. Before encrypted files are detected, identifying the DRM files

 doi:10.20944/preprints202601.1696.v1

is crucial using domestically implemented document security systems. Functionality improvements were required across all tools to enable custom signature analysis. DRM files help prevent the leakage of confidential data and intellectual property, and the evaluation results reflected the signature analysis employed for their identification and classification.

Table 22. DRM File Identification Results.

No.	Software	Tool					
		A	B	C	D	E*	F
1	Fasoo	-	-	-	-	-	-
2		-	-	-	-	-	-
3		-	-	-	-	-	-
4	Softcamp	-	-	-	-	O	-
5		-	-	-	-	-	-
6		-	-	-	-	-	-
7	Markany	-	-	-	-	O	-
8		-	-	-	-	O	-
9		-	-	-	-	O	-
10		-	-	-	-	O	-

4.2.5. Encrypted File Identification

The encrypted file identification performance of the tools was tested across eight file extensions, as listed in Table 23. Tool E performed best, followed by Tool B. The experiment included five document types and three compressed files encrypted using standard encryption. The identification performance of the models varied depending on the encryption algorithm employed. Not all the encrypted files used in the experiment were identified. Tool E successfully identified all the encrypted document-type files. Additionally, only Tools B and E identified encrypted files that were likely to contain sensitive information.

Table 23. DRM File Identification Results.

No.	File Type	Tool					
		A	B**	C	D	E*	F
1	doc	-	O	-	-	O	-
2	docx	-	O	O	-	O	-
3	hwp	-	-	-	-	O	-
4	ppt	-	-	-	-	O	-
5	pptx	-	O	O	-	O	-
6	7z	-	O	-	-	-	-
7	tar	-	-	-	-	-	-
8	zip	-	O	-	-	O	-

*; **.

 doi:10.20944/preprints202601.1696.v1

Table 24. File-preview Evaluation Results.

No.	Evaluation Item	File Type	Tool					
			A	B**	C*	D	E	F
1	Document Files	cell	-	-	O	-	-	-
2		csv	-	O	O	-	-	-
3		doc	-	O	O	-	-	-
4		docx	-	O	O	-	-	-
5		hwp	-	O	O	-	-	-
6		hwpx	-	-	-	-	-	-
7		pdf	-	O	O	-	-	-
8		ppt	-	O	O	-	-	-
9		pptx	-	O	O	-	-	-
10		rtf	-	O	O	-	-	-
11		show	-	O	O	-	-	-
12		txt	-	O	O	-	-	-
13		xls	-	O	O	O	-	-
14		xlsx	-	O	O	-	-	-
15	Compound Files	pst	-	O	O	O	O	-
16		ost	-	O	O	O	O	-
17		sqlite	-	O	-	-	O	-
18		db	-	O	-	-	O	-
19		7z	-	-	O	O	-	-
20		zip	-	-	O	O	-	-
21		Tar	-	-	O	O	-	-
22		egg	-	-	O	-	-	-
23		rar	-	-	O	O	-	-
24	Image Files (EXIF Support)	awd	-	-	O	-	O	-
25		psd	-	O	O	-	O	-
26		dwg	-	-	O	-	O	-
27		bmp	-	O	O exif	O exif	O	-
28		png	-	O exif	O exif	O exif	O exif	-
29		psp	-	-	O exif	O exif	-	-
30		jpeg	-	-	O exif	O exif	O exif	-
31		jpg	-	O exif	O exif	O exif	O exif	-

*; **.

 doi:10.20944/preprints202601.1696.v1

4.3. Seizure Phase

4.3.1. File-and Folder-Name Search

The collection and analysis performances of the tools were tested across seven evaluation items, as listed in Table 25. Tool E performed best, followed by Tool C. Based on the experimental results, logical image features exhibited different characteristics and formats for each tool. A key requirement for selective seizures is the ability to acquire logical images in a usable format and generate reports for the selected files; however, only Tool E offered this functionality. Tool D supported the VHD format, which enhanced its utility, but entailed a long acquisition time. The supported physical image format was E01 and there were no tool-supported memory-dump reports. Logical images focus on files and folders accessible to the user, allowing for the selective seizures of specific data. Additionally, they maintain the file system structure when copying data, thereby increasing their accessibility for analysis. Furthermore, to effectively handle exceptional cases, forensic tools must support memory dumps, physical image acquisition, and report generation.

Table 25. Collection Evaluation Result.

No.	Evaluation Item	Details	Tool					
			A	B	C*	D	E**	F
1	Logical Image	Image Format	-	-	O ctr	O vhd	O zip, dd	-
2		Report	-	-	-	O Save Error	O csv, txt	-
3		Time Taken for 200,000 Files	-	-	13 min	180 min	34 min	-
4	Memory Dump	Image Format	O bin	-	-	-	O raw	O mem
5		Report	-	-	-	-	-	-
6	Physical Image	Image Format	-	-	O	O	O	-
7		Report	-	-	O txt	O txt	O txt	-

*; **.

4.3.2. Extraction Evaluation

The extraction functionality of the tools was tested across the five items listed in Table 26. Tool E performed the best, followed by Tool B. All the tools supported report generation, with HTML being the most commonly supported format. In addition, all except one tool supported hash generation and hash sets. To complete the electronic evidence verification form, selective seizure tools must support these functions.

Table 26. Extraction Evaluation Results (Field Investigation Confirmation Form).

No.	Item	Tool					
		A	B**	C	D	E*	F
1	Report Support	O	O	O	O	O	O
2	Report Format	txt, html	html, excel,	html	txt	txt, csv, excel	html

 doi:10.20944/preprints202601.1696.v1

			text, kml, stix, tsk file				
3	Hash Support	-	O	O	O	O	O
4	Hash Set	-	O	O	O	O	O
5	Error Handling Report	O	O	O	-	-	O

**: The most recommended tool. **: Second recommended tool.

4.4. SSI Evaluation

It provides a comprehensive narrative explanation of how the Selective Seizure Capability Index was calculated in this study and why the SSI is a critical indicator for evaluating digital forensic tools in selective seizure scenarios. First, the SSI was designed to transform qualitative evaluation results into a reproducible and quantitative metric. In the experimental tables, each forensic tool was assessed using symbolic indicators O, □, △, and which represent different levels of functional support. To enable mathematical aggregation, these symbols were converted into numerical scores O corresponds to 3 points, □ to 2 points, △ to 1 point, and – to 0 points. This conversion establishes a consistent numerical basis while preserving the original qualitative meaning of the evaluation. Next, the evaluation was conducted separately for each investigative phase: Search, Select, and Seizure. Within each phase, multiple evaluation items were defined. For a given tool, the scores assigned to all items within a phase were summed to obtain the raw phase score. However, because the number of evaluation items differs across phases, direct comparison of raw scores would be inappropriate. Therefore, each phase score was normalized by dividing it by the maximum possible score for that phase, calculated as three times the number of evaluation items. This normalization process ensures that all phase scores fall within a 0–1 range and are directly comparable. After normalization, the phase scores were aggregated into a single SSI value using a weighted summation. In this study, weights of 0.3 were assigned to both the Search and Select phases, while a higher weight of 0.4 was assigned to the Seizure phase. This weighting scheme reflects the procedural importance of evidence acquisition, integrity verification, and reporting in determining evidentiary admissibility. The resulting SSI value therefore represents the overall degree to which a forensic tool supports selective seizure procedures across all investigative phases. Finally, for ease of interpretation, the SSI values were expressed as percentages by multiplying the normalized SSI by 100. For example, an SSI value of 0.71 indicates that the tool satisfies approximately 71% of the ideal selective seizure requirements defined by the evaluation model. The SSI is important for several reasons. First, it consolidates a large number of heterogeneous evaluation items into a single, interpretable indicator, enabling direct comparison between forensic tools. Second, it aligns tool evaluation with the procedural workflow of selective seizure, ensuring that performance is assessed in a manner consistent with real-world investigative practice. Third, the weighted structure of the SSI allows the evaluation to be adapted to different investigative priorities, such as emphasizing rapid on-site analysis or strict evidentiary admissibility. Finally, by providing a transparent and reproducible calculation process, the SSI enhances the objectivity and credibility of tool evaluation results, supporting both academic analysis and practical decision-making in digital forensic investigations. The Selective Seizure Capability Index provides substantial academic value in the performance evaluation of Windows forensic tools by advancing tool assessment from descriptive comparison to a structured, procedure-oriented evaluation methodology. Conventional studies on forensic tools often focus on the presence or absence of specific functions or on case-based effectiveness, which limits reproducibility and generalizability. In contrast, the SSI introduces a formalized mechanism for aggregating heterogeneous evaluation results into a coherent quantitative index, thereby

 doi:10.20944/preprints202601.1696.v1

strengthening methodological rigor. A primary contribution of the SSI lies in its ability to ensure fairness and comparability across evaluation dimensions. By converting qualitative assessment symbols into numerical scores and normalizing phase-level performance by their maximum attainable values, the SSI mitigates bias arising from unequal numbers of evaluation items across investigative phases. This normalization enables balanced comparison of tools even when functional coverage differs substantially, which is a common challenge in forensic tool evaluation. Moreover, the SSI explicitly aligns performance assessment with the procedural workflow of selective seizure, structured around the Search, Select, and Seizure phases. This phase-based aggregation reflects actual field investigation practices and distinguishes between analytical convenience and evidentiary completeness. As a result, tools are evaluated not merely on isolated technical features but on their capacity to support selective seizure as an integrated investigative procedure. The weighted aggregation employed in the SSI further enhances its academic relevance by allowing the evaluation model to reflect procedural priorities and legal considerations. Assigning a higher weight to the seizure phase emphasizes functions directly related to evidentiary integrity, verification, and reporting, which are critical for legal admissibility. At the same time, the weighting scheme remains adaptable, enabling researchers and practitioners to adjust phase importance according to investigative context or jurisdictional requirements without altering the underlying evaluation structure. From a normative perspective, the SSI also enables the quantification of principles that are traditionally addressed only at a conceptual level, such as proportionality and minimal intrusion. Performance in the selection phase directly influences the extent to which unnecessary data collection can be avoided. By incorporating these aspects into a measurable index, the SSI bridges the gap between legal and ethical objectives of selective seizure and the technical capabilities of forensic tools. Importantly, the SSI should not be interpreted solely as a ranking metric. Rather, it functions as an analytical framework that supports both comparative assessment and diagnostic interpretation. The decomposition of the SSI into phase-specific components allows researchers to identify strengths and limitations of tools at each procedural stage, facilitating targeted improvements and meaningful cross-study comparison. Consequently, the SSI establishes a reproducible and extensible foundation for future research on forensic tool verification and selective seizure evaluation in Windows-based environments

4.5. Discussion

This study proposed and applied an evaluation framework for selective seizure capabilities in Windows forensic tools, with a focus on live-response scenarios in investigative practice. While the framework provides a structured and practical basis for comparing forensic tools, several limitations should be acknowledged, which also indicate directions for future research. First, the experimental environment was limited to Windows 10 (version 22H2) using the NTFS file system, and the live system was emulated through a VMware-based virtual machine. Although this controlled environment enabled consistent comparisons across tools, the findings may not be directly generalizable to other operating systems, such as Windows 11, macOS, or Linux, nor to different file systems (e.g., APFS or EXT-based systems). Moreover, certain hardware-dependent conditions—such as TPM-based disk encryption, low-level I/O behaviors, or firmware-level artifacts—may not be fully reproduced in a virtualized environment. Future studies should therefore adopt a dual-validation approach that combines controlled virtual environments with physical hardware-based experiments to enhance external validity. Second, the scope of analysis focused primarily on file system artifacts and Windows event logs during the search and selection phases of selective seizure. Other increasingly important investigative domains, such as cloud storage services, instant messaging platforms, remote access tools, and containerized or virtualized execution environments, were intentionally excluded to maintain experimental clarity. As digital evidence environments continue to diversify, extending the proposed framework to cover these domains will be essential for maintaining its relevance in contemporary investigations. Third, the evaluation was conducted on a limited set of six portable, GUI-based forensic tools selected for their practical applicability in field

 doi:10.20944/preprints202601.1696.v1

investigations. To mitigate product bias, tool names and version identifiers were anonymized, and all tools were tested using their latest available versions at the time of evaluation. While this approach strengthened neutrality, it also constrained strict reproducibility by third parties. Future research may address this issue by documenting reproducible version indicators such as build numbers, module versions, or hash values without disclosing product identities, thereby balancing objectivity and replicability.

Finally, the present study did not explicitly incorporate usability and operational burden into the evaluation metrics. In real-world selective seizure scenarios, factors such as the number of procedural steps, required analyst expertise, degree of automation, and interpretability of outputs can significantly affect investigative efficiency and error risk. Incorporating usability-oriented indicators alongside functional capabilities would further enhance the framework's applicability to frontline investigative practice. Despite these limitations, this study contributes a structured and transparent evaluation framework that connects selective seizure functions with investigative efficiency and procedural compliance. By addressing the identified limitations, future research can extend the framework across platforms, environments, and investigative contexts, ultimately supporting more precise, proportionate, and legally robust digital evidence collection.

5. Conclusion

This study examined the importance of selective seizures based on digital evidence, and evaluated the functionality of six widely used tools to identify areas of improvement. First, the characteristics of digital evidence were used to evaluate the requirements for securing it, specifically, maintaining identity, reliability, and relevance to meet the admissibility criteria; then, the procedures, laws, and case precedents governing the selective seizure of evidence related to criminal activities in the field were reviewed. Subsequently, an evaluation model for selective seizure tools was developed encompassing key analysis items and a comprehensive dataset. Chapter 3 investigated new evaluation model analysis items for field response, reflecting the requirements covered in prior research on digital forensics tools. New field-response verification requirements were created based on existing requirement design indicators, and 6 types of digital forensics tools required for design and evaluation of verification requirement models were selected and datasets were developed. The results of the experiment, which involved three phases—search, select, and seize—revealed significant differences in the performances of the tools, allowing us to offer recommendations for improving their functionality. Specifically, in the search phase, Tools C and B outperformed NTFS parsing, whereas Tools C and B demonstrated superior capabilities in the Windows log analysis. In the selection phase, Tool C was most effective for file and folder name searches, Tool E for keyword searches, Tools B and C for file headers, Tool E for DRM and encrypted files, and Tool C for file previews. Tool E satisfied the evaluation criteria for data collection and extraction during the seizure phase. These findings offer valuable insight into the analysis and collection techniques of Windows forensic tools for field-based selective seizures. Therefore, they can assist forensic investigators in effectively analyzing and collecting digital evidence, thereby improving the efficiency and accuracy of their investigations. However, that study focused only on six portable tools implemented in an experimental environment comprising a virtual machine, an active system, Windows 10, and an NTFS. Future research should evaluate a wider range of tools geared toward other operating systems[46,47] such as Windows 11 and MacOS. Additionally, future research should explore the integration of artificial intelligence and machine learning techniques in selective seizure tools, particularly for cloud-based environments. AI-powered analytics could enhance the accuracy and efficiency of evidence identification in cloud storage, while also helping to address the challenges of data sovereignty and cross-jurisdictional investigations. The development of AI-assisted cloud forensic capabilities could provide investigators with more sophisticated tools for analyzing distributed digital evidence, though careful consideration must be given to maintaining the forensic integrity and admissibility of AI-processed evidence.

In addition, future investigations should consider incorporating various file systems and functions related to cloud storage[48–50], messaging applications, virtual environments, remote access programs, and third-party tools, all of which are being increasingly adopted. However, we expect that the findings of this study will contribute to the efficient acquisition of digital evidence using selective seizure tools, ultimately enhancing the capabilities of digital forensic investigations.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Sunho Kim; data collection: Cheolhee Yoon; analysis and interpretation of results: Cheolhee Yoon, Sunho Kim; draft manuscript preparation: Sunho Kim, Cheolhee Yoon. All authors re**viewed the results and approved the final version of the manuscript.**

Funding: This work was supported by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Ministry of Science and ICT (MSIT, Korea, No.RS-2024-00456709, A Development of Self-Evolving Deepfake Detection Technology to Prevent the Socially Malicious Use of Generative AI).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. G. O. Presidential Decree, Republic of Korea. (2023, Nov. 1). Regulations on Mutual Cooperation Between Prosecutors and Judicial Police Officers and General Investigative Principles, Article 41, Paragraphs 1, 2, 3 [Online]. Available: https://www.kicj.re.kr/board.es?mid=a20201000000&bid=0029&list_no=12687&act=view#wrap
2. Desktop Windows Version Market Share Republic of Korea. StatCounter Global Stats. Accessed on: Nov. 27, 2023. [Online]. Available: https://gs.statcounter.com/os-version-market-share/windows/desktop/south-korea
3. Computer Forensics Tools & Techniques Catalog - Tool Search. Accessed on: Dec. 16, 2023. [Online]. Available: https://toolcatalog.nist.gov/search/index.php
4. Desktop Hypervisor Solutions | VMware. Accessed on: Oct. 06, 2024. [Online]. Available: https://www.vmware.com/products/desktop-hypervisor/workstation-and-fusion
5. S. Watson and A. Dehghantanha, "Digital forensics: the missing piece of the cloud puzzle," Computer Fraud & Security, vol. 2016, no. 6, pp. 5-8, 2016.
6. R. Montasari et al., "Digital forensics: Challenges and opportunities for future studies," International Journal of Engineering & Technology, vol. 7, no. 2.28, pp. 125-135, 2019.
7. F. Nawaz et al., "Artificial intelligence-enabled digital forensics for cloud platforms," IEEE Access, vol. 9, pp. 129796-129813, 2021.
8. S. Yoon and S. Lee, "A study on digital evidence automatic screening system," J. Digit. Forensics KDFS, vol. 14, no. 3, pp. 239–251, Sep. 2020.
9. G. Cho, "A digital forensic analysis for directory in Windows file system," J. Korea Soc. Digit. Ind. Inf. Manag., vol. 11, no. 2, pp. 73–90, Jun. 2015.
10. Y. Shin, "A study on field detection techniques for non-searchable email attachments," M.S. thesis, Grad. Sch. Converg. Sci. Tech., Seoul Nat. Univ., Seoul, Korea, 2019.
11. T. -R. Lee and S. -U. Shin, "Reliability verification of evidence analysis tools for digital forensics," J. Korea Inst. Inf. Secur. Cryptol., vol. 21, no. 3, pp. 165–176, Jun. 2011.
12. J. Ham and I. J. Joshua, "A study on the comparison of modern digital forensic imaging software tools," J. Korean Inst. Internet Broadcast. Commun., vol. 19, no. 6, pp. 15–20, Dec. 2019.
13. M. Harbawi and A. Varol, "The role of artificial intelligence in digital forensics," in 2017 International Conference on Computer Science and Engineering (UBMK), pp. 110-115, IEEE, 2017.
14. C. Tassone et al., "Forensic examination of cloud storage services in Android," Digital Investigation, vol. 16, pp. 43-54, 2016.

 doi:10.20944/preprints202601.1696.v1

15. S. Park, G. Hur, and S. Lee, "Development of a set of data for verifying partition recovery tool and evaluation of recovery tool," J. Korea Inst. Inf. Secur. Cryptol., vol. 27, no. 6, pp. 1397–1404, Dec. 2017.
16. B. Martini and K.-K. R. Choo, "Cloud storage forensics: ownCloud as a case study," Digital Investigation, vol. 10, no. 4, pp. 287-299, 2013.
17. D. Quick and K. -K. R. Choo, "Big forensic data reduction: Digital forensic images and electronic evidence," Clust. Comput., vol. 19, no. 2, pp. 723–740, Jun. 2016.
18. A. Case et al., "Memory forensics: The path forward," Digital Investigation, vol. 20, pp. 23-33, 2017.
19. M. -S. Kim and S. Lee, "Development of Windows forensic tool for verifying a set of data," J. Korea Inst. Inf. Secur. Cryptol., vol. 25, no. 6, pp. 1421–1433, Dec. 2015.
20. G. Horsman, "Framework for reliable experimental design (FRED): A research framework to ensure the dependable interpretation of digital data for digital forensics," Computers & Security, vol. 73, pp. 294-306, 2018.
21. D. Lee, "A quality evaluation model based on ISO/IEC 9126 for digital forensic tools," M.S. thesis, Grad. Sch. Softw. Specialization, Soongsil Univ., Seoul, Korea, 2015.
22. M. Al-Saleh et al., "Cloud forensics: A research perspective," in 9th International Conference on Innovations in Information Technology, pp. 66-71, IEEE, 2013.
23. S. Alqahtany et al., "Forensic investigation of cloud storage services in the Internet of Things," Digital Investigation, vol. 29, pp. 1-10, 2019.
24. J. Park, J. R. Lyle, and B. Guttman, "Introduction to the NIST digital forensic tool verification system," Rev. KIISC, vol. 26, no. 5, pp. 54–61, Oct. 2016.
25. N. H. Ab Rahman and K.-K. R. Choo, "A survey of information security incident handling in the cloud," Computers & Security, vol. 49, pp. 45-69, 2015.
26. F. Daryabar et al., "Investigation of memory forensics of VMware workstation on Windows, Linux and Mac," Digital Investigation, vol. 22, pp. 566-578, 2017.
27. CFReDS Portal. Accessed on: Dec. 14, 2023. [Online]. Available: https://cfreds.nist.gov/
28. M. Damshenas et al., "Forensics investigation challenges in cloud computing environments," in 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic, pp. 190-194, IEEE, 2012.
29. K. Ruan et al., "Cloud forensics definitions and critical criteria for cloud forensic capability: An overview of survey results," Digital Investigation, vol. 10, no. 1, pp. 34-43, 2013.
30. Hosting various seminars related to digital forensics from 2001 to the present, DFRWS. Accessed on: Dec. 14, 2023. [Online]. Available: https://dfrws.org/
31. J. Dykstra and A. T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing," Digital Investigation, vol. 9, pp. S90-S98, 2012.
32. B. Martini and K.-K. R. Choo, "An integrated conceptual digital forensic framework for cloud computing," Digital Investigation, vol. 9, no. 2, pp. 71-80, 2012.
33. Forensic Focus. Accessed on: Dec. 13, 2023. [Online]. Available: https://www.forensicfocus.com/
34. T. Zia et al., "Digital forensics for cloud computing: A roadmap for future research," Digital Investigation, vol. 19, pp. 34-46, 2016
35. K. Koo et al., "Development of a mobile personal software platform," ETRI J., vol. 24, no. 4, pp. 31–32, 2009.
36. F. Cohen, "Digital Forensic Evidence Examination," in IEEE Security & Privacy, vol. 8, no. 2, pp. 68-71, 2020.
37. L. Chen et al., "AI-powered digital forensics: Opportunities, challenges, and future directions," IEEE Transactions on Information Forensics and Security, vol. 16, pp. 1-15, 2021.
38. D. Barrett and G. Kipper, "Virtualization and Forensics: A Digital Forensic Investigator's Guide to Virtual Environments," Syngress, 2010.
39. S. Raghavan, "Digital forensic research: current state of the art," CSI Transactions on ICT, vol. 1, no. 1, pp. 91-114, 2013.
40. A. Aminnezhad et al., "Cloud forensics issues and opportunities," International Journal of Information Processing and Management, vol. 4, no. 4, pp. 76-85, 2013.
41. R. McKemmish, "What is forensic computing?," Australian Institute of Criminology trends & issues in crime and criminal justice, no. 118, pp. 1-6, 2019.

 doi:10.20944/preprints202601.1696.v1

42. M. Taylor et al., "Digital evidence in cloud computing systems," Computer Law & Security Review, vol. 26, no. 3, pp. 304-308, 2010.
43. J. Park et al., "A Comparative Study of Digital Forensic Evidence Collection Methods for Cloud Storage Services," Digital Investigation, vol. 33, pp. 301-314, 2020.
44. S. Lin et al., "Artificial Intelligence for Digital Forensics: Challenges and Applications," IEEE Access, vol. 8, pp. 119697-119707, 2020.
45. K. Kim and S. Hong, "The Challenges of Cloud Forensics in South Korea," Journal of Digital Forensics, Security and Law, vol. 14, no. 4, pp. 45-57, 2019.
46. H. Chung et al., "Digital forensic investigation of cloud storage services," Digital Investigation, vol. 9, no. 2, pp. 81-95, 2012.
47. B. Lee et al., "Automated Forensic Analysis Framework for Cloud Computing Environment," Journal of Security Engineering, vol. 15, no. 6, pp. 435-444, 2018.
48. R. Adams et al., "The application of digital forensic readiness to cloud computing," in IFIP International Conference on Digital Forensics, pp. 47-56, Springer, 2013.
49. M. Jung and J. Kim, "Development of Digital Forensics Standard Model for Cloud Computing Services," Journal of the Korea Institute of Information Security & Cryptology, vol. 28, no. 2, pp. 403-413, 2018.
50. T. Yang et al., "Building Digital Forensic Investigation Framework for Cloud Computing," International Journal of Network Security, vol. 22, no. 6, pp. 978-985, 2020.