Forensic Science International: Digital Investigation 52 (2025) 301864

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

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

DFRWS EU 2025 - Selected Papers from the 12th Annual Digital Forensics Research Conference Europe

SOLVE-IT: A proposed digital forensic knowledge base inspired by MITRE ATT&CK

Christopher Hargreaves [a,b,*], Harm van Beek [c,d], Eoghan Casey [e]

[a] Department of Computer Science, University of Oxford, United Kingdom
[b] HARGS Solutions Ltd, Oxford, United Kingdom
[c] Netherlands Forensic Institute, The Hague, the Netherlands
[d] Department of Computer Science, Open Universiteit, Heerlen, the Netherlands
[e] Ecole des Sciences Criminelles, University of Lausanne, Batochime, 1015, Lausanne, Switzerland

ARTICLE INFO

Keywords:
Digital forensic techniques
Digital forensic science
Quality assurance
Error-focused datasets
AI applications

ABSTRACT

This work presents SOLVE-IT (Systematic Objective-based Listing of Various Established (Digital) Investigation Techniques), a digital forensics knowledge base inspired by the MITRE ATT&CK cybersecurity resource. Several applications of the knowledge-base are demonstrated: strengthening tool testing by scoping error-focused data sets for a technique, reinforcing digital forensic techniques by cataloguing available mitigations for weaknesses (a systematic approach to performing Error Mitigation Analysis), bolstering quality assurance by identifying potential weaknesses in a specific digital forensic investigation or standard processes, structured consideration of potential uses of AI in digital forensics, augmenting automation by highlighting relevant CASE ontology classes and identifying ontology gaps, and prioritizing innovation by identifying academic research opportunities. The paper provides the structure and partial implementation of a knowledge base that includes an organised set of 104 digital forensic techniques, organised over 17 objectives, with detailed descriptions, errors, and mitigations provided for 33 of them. The knowledge base is hosted on an open platform (GitHub) to allow crowdsourced contributions to evolve the contents. Tools are also provided to export the machine readable back-end data into usable formats such as spreadsheets to support many applications, including systematic error mitigation and quality assurance documentation.

1. Introduction

The growing awareness of weaknesses in digital forensic investigations has expanded requirements for quality assurance. Missed, mishandled, and misinterpreted digital evidence can result in it being dismissed, scarce resources being wasted, and trust in digital evidence being diminished. To reduce these risks, standards bodies are promoting systematic approaches to mitigate errors and uncertainty in digital evidence, but the field lacks a practical solution to implement these requirements.

Many digital forensic process models have been proposed at different levels of abstraction that offer advantages for teaching and discussion at a high-level. However, other fields that need to model specific technical approaches have adopted a complementary knowledge-base approach e.g. MITRE ATT&CK, and D3FEND in cybersecurity (Strom et al., 2018; MITRE, 2024a,c).

This paper demonstrates that a similar knowledge base in digital forensics could have many benefits in areas such as creating error-focused data sets, quality assurance, and systematic mitigation of errors and uncertainty. Documenting processes in more detail than high-level topic names can provide solutions in these areas.

A recent paper (Hargreaves et al., 2024b) provided a deconstruction of 'internal processes' within monolithic digital forensic tools and showed the dependencies between them, along with examples of errors that can occur at each stage. This paper extends that recent paper and considers the overall digital forensic process rather than just monolithic digital forensic tools. It takes a bottom-up approach to mitigating weaknesses within all aspects of a digital forensic investigation, taking a technique-based view of digital forensics. Incorporating weaknesses and mitigations, and providing tools for applying the knowledge base to investigations, provides a practical solution called for in ASTM E3016-18, the Standard Guide for Establishing Confidence in Digital and

* Corresponding author. Department of Computer Science, University of Oxford, United Kingdom.
E-mail addresses: christopher.hargreaves@cs.ox.ac.uk (C. Hargreaves), harm.van.beek@nfi.nl (H. van Beek), eoghan.casey@unil.ch (E. Casey).

https://doi.org/10.1016/j.fsidi.2025.301864

Available online 24 March 2025

Multimedia Evidence Forensic Results by Error Mitigation Analysis (ASTM, 2018). Error mitigation in a given context requires two things: 1) a comprehensive catalogue of potential sources of error, and 2) methods to mitigate each of the potential errors. This work facilitates both of these.

Due to the success and broad applications of MITRE ATT&CK and D3FEND knowledge bases in cybersecurity, they are used as inspiration, and this paper presents the SOLVE-IT knowledge base, provides a starting point for cataloguing the equivalent 'tactics' and 'techniques' in ATT&CK that can be used during digital forensic investigations. Opening this knowledge base to the broader community enables crowdsourcing, to curate collective knowledge of weaknesses and mitigations, to keep pace with advances in digital forensic methods.

The paper makes the following contributions:

- A knowledge base modelled on MITRE ATT&CK, indexing 104 digital forensic techniques across 17 categories, with detailed descriptions, potential weaknesses, mitigations, examples, and references for 33 of them. In total 156 weaknesses and 108 mitigations are indexed.
- A central Github repository hosting the knowledge base, facilitating crowdsourced effort in expanding and enhancing the resource, evidenced through contributions to this initial version from other researchers in the community.
- Demonstrative examples of the use of this knowledge base, covering a variety of applications: building error-focused datasets, reflecting on an investigation to systematically consider potential weaknesses, integration with the CASE ontology (CASE, 2024), and an example of a systematic review of AI uses in digital forensics.

The remainder of this paper is structured as follows: Section 2 provides a discussion of background and related work. Section 3 discusses the approach for developing the knowledge base, followed by an introduction to its structure in Section 4. Examples of the use of the knowledge base are presented in Section 5 and 6, followed by a discussion of limitations and further work in Section 7, and conclusions in Section 8.

2. Background and related work

Many process models and frameworks have been proposed for digital forensic investigations, taking a top-down approach to structure the overall process. Horsman and Sunde (2022) provides references to 11 digital forensic investigation processes, 11 best practise guidelines, and 14 empirical studies of digital forensic practise. While there are applications of some of these frameworks, they rarely focus on technical details and as a result do not provide practical ways to mitigate errors that can occur during a digital forensic investigation, which is a major ongoing challenge.

2.1. Consideration of error

The increasing quantity, complexity, and diversity of continuously evolving digital technologies and forensic tools and techniques make it difficult for practitioners and developers to stay up-to-date, let alone know about every potential problem that could impact the evidential value of digital traces. Identifying associated mitigations is also difficult, and even when they are identified in research, according to the DFPulse 2024 Practitioner Survey (Hargreaves et al., 2024a), academic work is not transitioning into the hands of practitioners. Sunde (2022b) also investigated how objectivity and reliability of evidence is approached by practitioners and reported that a third "did not use any technique for maintaining examiner objectivity".

Ryser (2024) thoroughly compiles sources of weaknesses, providing a high-level map of the current issues encompassing errors in processes, tools, and expertise. Sunde (2022a) calls for "designing adequate measures for auditability, error minimisation and sustaining the necessary quality of the procedures and the results" and raises doubts about proposed solutions that concentrate on a single phase of digital forensic investigations because the influence of cognitive and human factors are so pervasive throughout the process. For instance, peer review of digital forensic findings and conclusions cannot resolve the problem of practitioners overlooking relevant digital traces during earlier phases, or tool errors missing or misinterpreting evidence. Collie (2018) describes that these challenges caused a crisis in the UK, resulting in stricter ISO/IEC 17025 accreditation requirements for digital forensic organisations. While Cusack and Homewood (2013), Ryser (2024), Horsman (2024), and Hargreaves et al. (2024b) have highlighted weaknesses in digital forensic investigations, none have provided a systematic solution for mitigating those weaknesses.

2.2. Subdomains, ontologies and techniques

There is also work categorising subdomains of digital forensics. Karie and Venter (2014) describes subfields of digital forensics: computer, software, database, multimedia, device, and network forensics, which was used by the Netherlands Register of Court Experts (NRGD) to define its fields of expertise. An extended version is presented in Wu et al. (2020), merging some categories and adding separate categories for memory and malware. Al-Dhaqm et al. (2021) is similar and provides further subdomains, e.g. Small Device and Subsystems includes: Drone Forensics, IoT forensics, and Appliance Forensics.

Modelling also exists of concepts within digital forensics via the Cyber-investigation Analysis Standard Expression (CASE) Ontology (Casey et al., 2017; CASE, 2024). It is focused on 'Observable Objects' and their facets. Examples include 'observable:NTFSFile' and 'observable:keywordSearchTerm'. CASE also has the concept of 'InvestigativeAction', which covers actions applied to digital evidence to examine or analyse data. However, while CASE provides a set of foundational cyber-investigation classes, it defers more specialized classes to downstream, adopting models. Integration of SOLVE-IT with CASE is discussed later in Section 5.4.

There is also work considering specific tasks within digital forensics. ISO/IEC (2015) 27042:2015 discusses that an investigation is made up of examinations leading to interpretations, resulting in reports. Those examinations are composed of several analyses, where each analysis is composed of several validated processes. It also discusses interpretation and reporting in more detail, but does not detail specific processes to be applied. NIST (2022) states "There are hundreds if not thousands of individual techniques that might be employed in a digital forensic examination" but presents seven high-level categories from 'Protecting Data During Acquisition', through to 'Analysis of Results', but also subtechniques within these, for example: 'Acquisition of Digital Data' includes: 'Storage Device (Hard Drive & Flash Drive) Data Acquisition', 'Mobile Device Acquisition', 'Other Device Data Acquisition', and 'Social Media Data Acquisition'. It provides a description of these techniques, but does not standardise their representations. Another model (van Beek, 2018) presents five stages of a digital investigation ('Recognize & Preserve', 'Acquire', 'Extract', 'Relate', 'Evaluate'), based on the steps that provide actual output and describes that methods, tools, and procedures can be mapped to each stage. Finally in terms of modelling detailed techniques within tools, Hargreaves et al. (2024b) recently modelled an 'abstract forensic tool' based on the inferred inner workings of Autopsy, Axiom and X-Ways Forensics. That work began to examine the digital forensic process in more detail but was focused on the perspective of the processes performed inside a monolithic digital forensic tool.

2.3. Related catalogues of techniques

The MITRE ATT&CK knowledge base (MITRE, 2024b), and the associated D3FEND knowledge base (MITRE, 2024c) have been very

successful in consolidating work in the cybersecurity area. ATT&CK is described as a "knowledge base of adversarial techniques based on real-world observations. ATT&CK focuses on how adversaries interact with systems during an operation, reflecting the various phases of an adversary's attack lifecycle and the platforms they are known to target." This approach and knowledge base has been shown to have significant applications, described in ATT&CK resources (e.g. detections and analytics, threat intelligence, adversary emulation and red teaming, and assessment and engineering), but also in academic literature e.g. Ahmed et al. (2022). There are several key concepts within ATT&CK:

- **Tactics** - "Tactics represent the 'why' of an ATT&CK technique or sub-technique. It is the adversary's tactical goal: the reason for performing an action. For example, an adversary may want to achieve credential access."
- **Techniques** - "Techniques represent 'how' an adversary achieves a tactical goal by performing an action. For example, an adversary may dump credentials to achieve credential access."
- **Sub-techniques** - "a more specific or lower-level description of adversarial behavior"
- **Procedures** - "specific implementation or in-the-wild use the adversary uses for techniques or sub-techniques"

These are then organised as matrices with tactics along the top, and techniques represented under each of the tactics headings. There are different matrices for enterprise, mobile, and Industrial Control Systems (ICS). An example section of the Enterprise Matrix is shown in Fig. 1.

For each ATT&CK technique, the knowledge base includes:

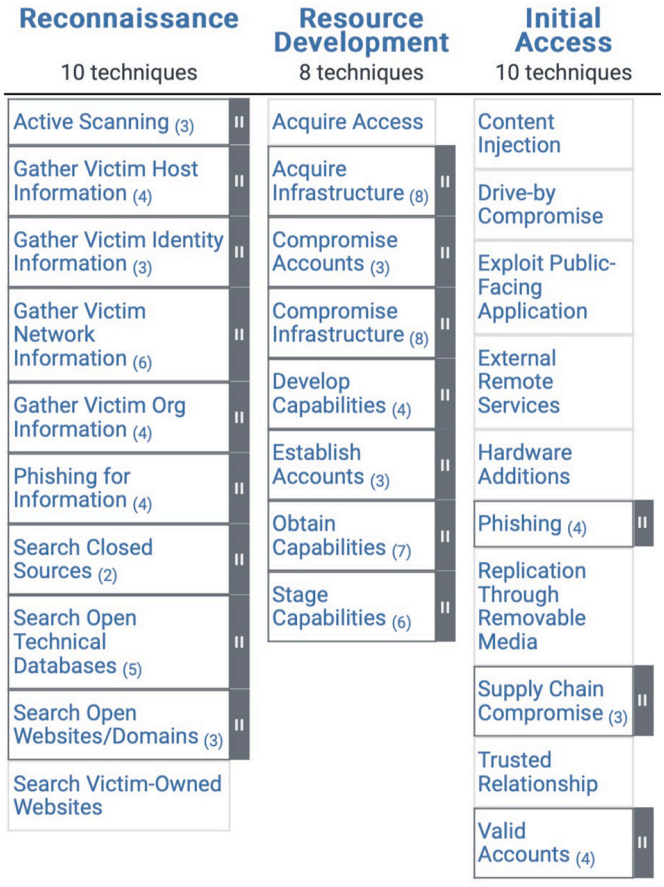

Fig. 1. Screenshot showing an extract from MITRE ATT&CK knowledge base showing 3/14 tactics on the top, and techniques below each of those tactics (MITRE, 2024a).

- A description of the technique;
- A list of sub-techniques related to the technique;
- Procedure examples;
- A list of known mitigation methods for the technique;
- A list of known detection methods for the technique;
- Related references and additional resources;

Other details are also preserved such as the current version of the technique descriptor, the contributors, when it was created and last modified, and what platforms it applies to.

The corresponding D3FEND knowledge base is: "A knowledge graph of cybersecurity countermeasures" and contains a series of tactics and techniques but from the defender's perspective. This is linked to ATT&CK and has some investigative aspects (in the 'Defend' subsection) but is focused on preventing, detecting, and investigating cyber incidents specifically rather than having a digital forensics focus. Nevertheless, the fields used in each of the techniques described are: Definition; How it works; Digital Artefact Relationships; Related ATT&CK Techniques; and References.

2.4. Summary

There are many existing process models and frameworks proposed for digital forensics. However, if the parallel area of cybersecurity is considered, it is apparent that none of the existing digital forensic models achieve the level of detail and practicality of the MITRE ATT&CK knowledge base. It is hypothesised that digital forensics would benefit from such a comprehensive, structured knowledge base, mirroring the tactics and techniques in ATT&CK, but focused on digital forensic investigation techniques.

3. Methodology

3.1. Overall aim and approach

The long-term aim of this research is to create a comparable knowledge base to ATT&CK for digital forensic tactics and techniques. The first public ATT&CK model was released in May 2015 (96 techniques under 9 tactics) (Strom et al., 2018), and at time of writing is at version 16.1 over 9 years later, with 657 techniques/sub-techniques in the enterprise matrix alone. It is infeasible to match the level of maturity in a single iteration in a single paper, so this work aims specifically to create a viable initial version embedded in infrastructure such that the digital forensic community has a starting point for consulting, interacting with, and improving the content of the knowledge base. It also demonstrates the value of such a resource to the community through several illustrated examples. This work primarily makes reference to ATT&CK as the more mature resource, but D3FEND was also consulted while designing SOLVE-IT.

This work does not conduct or report on a systematic literature review. Instead it focuses on the structure of the knowledge base and demonstrating its value and applications, rather than comprehensive population and organisation of the knowledge base, which is intended to be a community effort. The content is intended to be enough to showcase the value of the resource. Nevertheless, this has resulted in the identification of 104 techniques, 156 weaknesses, and 108 mitigations so far.

To populate a subset of the knowledge base, some existing framework publications were considered. The review of 10 years of DFRWS EU publications (Breitinger et al., 2024) categorised 27 papers as 'techniques/fundamentals' which were reviewed and some qualified as techniques for this work. Also the recent DFPulse 2024 Practitioner Survey (Hargreaves et al., 2024a) asked how often 20 different techniques were used in practise, and these were added to the knowledge base if they detailed something tangible that could be carried out as a process, or a practical resource could be found that referenced them. In addition a Google Scholar search was conducted for 'digital forensic

technique' and several were added based on the top 50 results, specifically those that had tangible distinct approaches. Finally, the authors' experience and awareness of methods in investigations was also used in identifying techniques.

3.2. Knowledge transfer from MITRE ATT&CK

This section describes how the knowledge from MITRE ATT&CK and D3FEND is mapped to digital forensics.

3.2.1. Tactics (Objectives) and techniques

Reiterating that in ATT&CK, tactics represent "the adversary's tactical goal: the reason for performing an action. For example, an adversary may want to achieve credential access." Tactics are the column headings within the knowledge base.

Translating this to digital forensics, tactics are rephrased as objectives, and represent a (sub)goal in a digital investigation. Despite the renaming, the same idea persists.

Definition 1. objectives are "the goal that one might wish to achieve in a digital forensic investigation", e.g. acquire data, or extract information from a file system.

In ATT&CK, techniques represent "how an adversary achieves a tactical goal by performing an action. For example, an adversary may dump credentials to achieve credential access." In this work, for digital forensics, techniques are the entries in each of the columns, and can be defined as:

Definition 2. techniques represent: "how one might achieve an objective in digital forensics by performing an action". e.g. for the objective of 'acquire data', the technique 'disk imaging' could be used.

Techniques here can also be mapped to the CASE Ontology. In some cases, the class *investigation:InvestigativeAction* (CASE Ontology, 2024c) can be used, which is defined as "something that may be done or performed within the context of an investigation, typically to examine or analyse evidence or other data". This closely maps to techniques in SOLVE-IT, and some techniques e.g. disk imaging, are included in specific CASE examples (CASE Ontology, 2024b). However, in some cases where multiple actions are taken as part of a technique, the *action:ActionLifecycle* (CASE Ontology, 2024a) may be more appropriate. Additional mapping of CASE classes is discussed in Section 5.4. Despite these parallels, while CASE can support a knowledge base of techniques, it is not one itself.

In terms of the organisation of the techniques, Beebe and Clark (2005) describe that frameworks should be based on objectives rather than tasks, as a different subset of steps is needed in each situation. Within SOLVE-IT, this approach is adopted. Not all objectives and not all techniques from the knowledge base are needed in all cases, so the objectives do not represent a linear, complete, or definitive sequence. This is illustrated later in Section 5. However, the objective set selected for this initial version to organise the techniques does not follow the phases presented in Beebe and Clark (2005). It more closely follows the more recent digital forensic tool phases in Hargreaves et al. (2024b) (e.g. 'extract partition and file system information'), with additions for parts of a digital investigation that are not included by the previous dissection of digital forensic *analysis* tools, e.g. acquisition, research, etc. It is also shown in Section 4 how the techniques can be reorganised according to any framework, set of phases, or other criteria desired, so establishment of a single 'perfect' model is unnecessary.

3.2.2. Information within techniques

Section 2 discussed the information contained within techniques in the ATT&CK and D3FEND knowledge bases. Based on that structure, and also including information regarding potential errors in digital forensic tools from Hargreaves et al. (2024b), the structure of a technique is as follows:

- **ID** - The technique's ID, e.g. T1001;
- **Technique name** - The name of the technique;
- **Technique description** - A short description of what the technique involves;
- **Synonyms** - Any possible synonyms for the technique;
- **Details** - Further details beyond the short description;
- **Subtechniques** - Some techniques may have subtechniques, as per ATT&CK techniques and in NIST (2022) and can be listed here, referenced by technique ID;
- **Examples** - Examples related to the technique. These can be datasets that use the techniques, example cases that made use of the techniques either from published cases or synthetic ones, or specific tools that provide the technique;
- **Weaknesses** - Using the structured approach to considering error presented in Hargreaves et al. (2024b) based on error classes in ASTM (2018), these are reframed as 'weaknesses'. This field allows potential weaknesses associated with techniques to be referenced, pointing to indexed weaknesses within the knowledge base (see below);
- **CASE output entities** - Any potential CASE Ontology entities that allow the technique output to be represented. This is discussed further in Section 5.4.
- **References** - References can and should be included to support definitions and examples for the techniques as "[it is] methods, based on science, which form the foundation for forensic activities" (van Beek, 2018).

3.2.3. Information within weaknesses

The structure of the weaknesses is as follows:

- **ID** - The weakness's ID (e.g. W1001);
- **Description** - A short description of the weakness;
- **INCOMP** - Results in incompleteness;
- **INAC-EX** - Results in inaccuracy:existence;
- **INAC-AS** - Results in inaccuracy:association;
- **INAC-ALT** - Results in inaccuracy:alteration;
- **INAC-COR** - Results in inaccuracy:corruption;
- **MISINT** - Results in potential misinterpretation;
- **Mitigations** - Building on Hargreaves et al. (2024b) which discussed errors, this work also provides indexed references to any mitigations that could minimise or reduce the impact of individual weaknesses (see Section 3.2.4);
- **References** - These should be included to support definitions and examples, including to error-focused datasets demonstrating the weakness (see Section 5.1).

3.2.4. Information within mitigations

The structure of the mitigations is as follows:

- **ID** - The mitigation's ID (e.g. M1001);
- **Mitigation name** - A short description of the mitigation;
- **Details** - A longer description for the mitigation;
- **Technique** - An optional index to a related technique. This can be used when a mitigation is sufficiently complex to be considered a technique in its own right (see example in Section 4.1);
- **References** - These should be included to support the description of the mitigation.

3.3. Representation and implementation

The knowledge base is currently hosted on Github,[1] where each technique, weakness, and mitigation is represented in JSON form. At present, manual code edits are needed to update the content, but in

[1] https://github.com/SOLVE-IT-DF/solve-it

future this data could form part of the backend of an interactive tool (much like the crowdsourced artefact catalogue in Casey et al. (2022)) for searching, reviewing, and updating the knowledge base. Two utility Python scripts are provided that generate spreadsheets from the JSON data for ease of review. The first creates the overall view shown in Fig. 2. Each of the techniques listed is a link within the spreadsheet to the details of that technique as a separate worksheet, such as those shown in Figs. 3 and 4. The second Python script generates a different representation of the weaknesses and mitigations that can be used to review an investigation from a quality assurance perspective (discussed in Section 5.3).

4. Examples from the knowledge base

Using the information derived from the sources described in the previous section, and through an iterative process of thematic grouping, the model was created. At present, the model contains 104 techniques organised over 17 objectives.

Due to space constraints, each objective and technique cannot be discussed in detail within this paper. However, an overview of the knowledge base is shown in Fig. 2 and illustrative examples of the techniques are discussed in Sections 4.1 and 4.2, along with the demonstrations of the use of the knowledge base in Section 5.

It is important to note that the categorisation of the techniques (objectives in this case) shown in Fig. 2 is a presentation and organisational convenience, and the data model supports organisation of the techniques based on any process model, taxonomy or other system that is needed. A JSON configuration file can provide the high-level categories along with the techniques that should be included within them. In addition to the primary *solve-it.json* categorisation, configuration files are provided for the Carrier et al. (2003) model (acquisition, analysis, presentation) and the DFRWS/Palmer model (Palmer et al., 2001) (preservation, collection, validation, identification, analysis, interpretation, documentation, and presentation). These demonstrate the flexibility of the knowledge base but also illustrate that when large numbers of techniques are indexed, higher granularity of the overarching categorisations is needed. The following subsections describe two of the techniques in the knowledge base as illustrations.

4.1. Example technique 1: disk imaging

As the first example, Fig. 3 in the appendix shows the latest version of the *T1002:Disk Imaging* technique within the *'Acquire'* objective. It shows there are nine weaknesses identified within this technique so far. Each of these has been given a unique ID. In addition, each weakness is mapped against the resulting 'error' classifications: INCOMP, INAC-EX, INAC-AS, INAC-ALT, INAC-COR, MSINT, and where possible, mitigations to the potential weakness are also linked.

For example: the weakness *W1006: Acquisition does not include data in HPA*, has an associated mitigation of: *M1005: Testing to ensure software and hardware setup detects HPAs*. In addition, *W1014: Imaging process changes original data* has two associated mitigations of: *M1007:Use hardware write blocker* and *M1008: Use software write blocker*. These two mitigations are complex enough that they link through to additional techniques within the knowledge base (*T1012:Hardware Write Blockers*, and *T1013:Software Write Blockers*), which are also shown in Fig. 2. Linking to a technique is not required, but is possible for any mitigation. Also note that specific references and resources indexed in the knowledge base are mapped against either the general technique, or against specific weaknesses or mitigations. Finally, a CASE output class identified for this technique is 'observable:Image' ("a complete copy of a hard disk, memory, or other digital media." (CASE Ontology, 2024d)).

4.2. Example technique 2: review content for relevant material

As a second example, Fig. 4 in the appendix shows the technique *T1054:Content review for relevant material*, within the *'Locate Relevant Digital Artefacts'* objective. This illustrates that it is possible to represent

Survey	Preserve	Prioritise	Acquire	Gain Access	Process Storage Format	Perform Data Reduction	Locate Relevant Digital Artefacts	Extract Partition and File System Information	Extract Operating System Feature Information	Extract Application-based Information	Examine data at the file-level	Establish Identities	Visualisation	Event Reconstruction	Research	Reporting
Crime scene searching T1005	Place device in faraday environment T1010	Triage T1001	Disk imaging T1002	Key recovery from memory T1031	Disk image hash verification T1042	Privileged material protection T1046	Keyword searching T1049	Identify partitions T1059	Content indexer examination (OS) T1065	Browser examination T1069	Audio content analysis T1079	Extraction of user accounts T1084	Virtualise suspect system for previewing T1103	Timeline analysis T1086	Source code review T1089	Bookmarking T1091
Digital sniffer dogs T1006	Evidence bags T1011		Memory imaging T1003	Side channel T1032	Forensic image format decoding T1043	Hash matching (reduce) T1047	Hash matching (locate) T1050	Process file system structures T1060	Log file examination (OS) T1066	Email examination T1070	Video content analysis T1080	Identify conflation T1085		Geospatial analysis T1087	Experimentation T1090	Produce bookmark-based automated report T1092
SyncTriage T1007	Hardware write blockers T1012		Selective data acquisition T1004	Extraction of account details from an accessible device T1033	Mobile backup decoding T1044	Privacy protection via partial processing T1048	Fuzzy hash matching T1051	Non-allocated file recovery T1061	Cloud synchronisation feature examination (OS) T1067	Database examination T1071	Image content analysis T1081			Connection analysis T1088	Instrumentation T1095	Write expert report T1093
Profiling network traffic T1008	Software write blockers T1013		Privacy preserving selective extraction T1015	Brute force attack T1034	Decode standard archive format T1045		Timeline generation T1052	Decryption of encrypted file systems/volumes T1062	Recently used file identification (OS) T1068	Chat app examination T1072	Document content analysis T1082				Cell site survey T1101	Disclosure T1094
Locate cloud account identifiers T1009	Chain of custody documentation T1014		Live data collection T1016	Dictionary attack T1035	Decode data from image from unmanaged NAND T1102		Entity extraction T1053	Identify file types T1063	Memory examination (OS-level) T1083	Calendar app examination T1073	File repair with grafting T1099					
			Network packet capture T1017	Smudge attack T1036			Content review for relevant material T1054	File carving T1064	Run programs identification (OS) T1096	Social network app examination T1074	EXIF data examination T1100					
			Remote data collection T1018	Obtain password from suspect T1037			File system content inspection T1055		Installed programs identification (OS) T1097	Mapping app examination T1075						
			Mobile backup extraction T1019	Rainbow tables T1038			Entity connection identification T1056		User account analysis (OS) T1098	Photos app examination T1077						
			Mobile file system extraction T1020	App downgrade T1039			Steganography detection T1057			Cloud sync app examination T1078						
			Mobile storage imaging T1021	Use mobile device exploit T1040			Mismatched file extension detection T1058			Memory examination (application-level) T1105						
			Mobile device screenshot based capture T1022	Pin2Pwn T1041												
			Cloud data collection using account details T1023													
			Cloud data collection via request T1024													
			Writing data to a forensic image format T1025													
			Writing data in standard archive format T1026													
			Data read using JTAG T1027													
			Chip-off T1028													
			Data read from desoldered eMMC T1029													
			Data read from unmanaged NAND T1030													
			Collect data using open source intelligence T1104													

Fig. 2. Overview of SOLVE-IT model, viewed as an exported spreadsheet (produced with *generate_excel_from_kb.py* from the backend JSON data). Shows 104 identified techniques, over 17 categories, with 33 populated with details (shaded in grey) indicating that some weaknesses and mitigations have been documented.

not just software weaknesses, but also human ones, and not just software mitigations, but also procedures.

There are currently three potential weaknesses for this technique: *W1060: a relevant piece of media was not flagged as relevant*, *W1061: An irrelevant piece of media was flagged as relevant*, and *W1062: The grade or specific tag given to an item is incorrect*. Each of these are mapped to various mitigations shown in the figure, including *M1039:Hash matching to reduce need for manual review*, *M1040:Use of AI to flag potentially relevant content to reduce need for manual review*, *M1041:Regular breaks to minimise investigator fatigue*, and *M1042:Secondary review of flagged items to ensure relevance*.

This also demonstrates the use of sub-techniques where the following techniques are referenced: *T0179:Audio content analysis*, *T1080:Video content analysis*, *T1081:Image content analysis* and *T1082:Document content analysis*.

5. Demonstrative examples

As there are multiple existing models of digital forensic investigations it is important to demonstrate utility from additional work in this area. Importantly, this is not a pure framework or process model, and the knowledge base provides additional specific information to enable several practical uses.

The remainder of this section provides six demonstrations that are facilitated by the knowledge base that cannot be easily achieved using existing work: scoping error-focused data sets (Section 5.1), cataloguing mitigations for specific weaknesses (Section 5.2), generating worksheets to support error mitigation analysis and producing quality assurance documentation (Section 5.3). It also provides a new interface to the CASE Ontology and can identify gaps (Section 5.4), considers the potential use of AI in digital forensics in a structured manner (Section 5.5), and is used to identify areas for future academic work (Section 5.6). Several other uses are given later in Section 7.2, though without examples due to space constraints.

5.1. Scoping error-focused data sets

The first application of the knowledge base is that for many of the potential weaknesses it can highlight what is needed from an error-focused data set.

In Hargreaves et al. (2024b), an error-focused dataset is provided along with a CASE representation of the ground truth. The knowledge base has been updated to index this dataset, related to the technique *T1059:Identify Partitions*, and linked to *W1065: Missing deleted but recoverable partitions*. Associated mitigations have also been indexed (*M1043: Scan for orphaned EPTs*, *M1044: Scan for VBRs in unpartitioned space*), and are linked to reference tool features such as 'Scan for lost partitions' in X-Ways Forensics. It is also indexed in *T1060:Non-allocated file recovery* as the dataset also includes an example of *W1135:File metadata and name of a non-allocated file are recovered, but its content has been overwritten by a newer file, but the non-allocated file's content is presented as this newer file's data*.

While the knowledge base can be used to index existing datasets, it can also be used to identify that new ones should be created. Considering one of the example techniques: *T1072:Chat app examination*, it is possible to review the possible weaknesses and consider what is needed from datasets to test whether the tools in use will minimise or avoid the weakness's results. The potential weaknesses included for *T1072* are shown in Table 1. Datasets can then be constructed to ensure an understanding of the tool behaviour in these use cases is known.

5.1.1. iOS messages example

As a first example, using the weaknesses in Table 1, the weakness *W1090:Recovering a live message with incorrect metadata* and *W1092: Recovering a non-allocated message with incorrect metadata* can be considered with regard to iOS Messages database changes between iOS 10 and iOS 11. Here the timestamp format was changed in sms.db to use a higher resolution, but only for new messages. This resulted in some tools initially not parsing this correctly (Barnhart, 2017). This weakness leads to the linked mitigations shown in Table 2.

Table 1
Weaknesses in *T1072:Chat app examination*, motivating the creation of specific error-focused datasets.

ID	Weakness
W1085	Missing messages from the live set of messages
W1086	Failing to recover non-allocated but recoverable messages
W1087	Presenting a live message that did not exist
W1088	Presenting a deleted message that did not exist
W1089	Recovering a live message with incorrect content
W1090	Recovering a live message with incorrect metadata
W1091	Recovering a non-allocated message with incorrect content
W1092	Recovering a non-allocated message with incorrect metadata
W1093	Presenting a deleted message as live
W1094	Attributing a message to the incorrect sender
W1095	Attributing a message to the incorrect thread
W1096	Failing to recover attachments for a live message
W1097	Failing to recover attachment for a non-allocated message
W1098	Assigning incorrect metadata to a message attachment
W1099	Assigning an attachment to an incorrect messages
W1100	Failure to display special effects or highlight within a message
W1101	Failure to recover message edits if available
W1102	Failure to display that a message had a previous state

Having a clear weakness associated with a specific technique allows the need for a dataset to be identified, one created, and indexed within the knowledge base and therefore linked directly with particular problems that can occur during an examination.

5.1.2. WhatsApp example

A second example from BinaryHick (2022) discusses the changes identified in Whatsapp version 2.22.11.82, where the *messages* table was changed to *message* (singular). Failure to take into account this change would result in *W1085:Missing messages from the live set of messages*, and also reports that attachments may not be processed correctly, therefore resulting in *W1096:Failing to recover attachments for a live message*. Future reports of examples of this nature would ideally produce error-focused datasets capturing the problem, and they can be indexed against a specific weakness in the knowledge base.

Since operating systems and apps are often updated, This example shows that even validated methods might give wrong results if applied to newer case data. In contrast to forensic methods applied in the physical realm, implementations of digital forensic techniques require continuous updates to remain valid.

5.2. Highlighting available mitigations for specific errors

In addition to error-focused datasets in the previous section which are most relevant for tool testing programmes, SOLVE-IT can also help practitioners locate specific mitigations that are available when they encounter or become aware of a specific weakness when using a technique. This is a particularly important application because as highlighted in the recent *DFPulse 2024 Practitioner Survey* (Hargreaves et al., 2024a), the visibility of academic work to practitioners is very poor. The SOLVE-IT knowledge base references academic work that offers

Table 2
Mitigations for W1090.

ID	Mitigation
M1027	Dual tool verification
M1050	Manual verification of relevant data
M1054	Testing of message extraction and parsing features
M1055	Correlation of message data with service provider
M1056	Correlation of message data from third party data (another participant in message thread)

potential practical benefits via mitigations, or work that highlights weaknesses in approaches, making the referenced work more visible and associated with a specific practical technique that is in use. Two examples follow, one continuing with iOS messages (5.2.1), and a second related to file carving (5.2.2).

5.2.1. iOS messages

For weakness *W1086: Failing to recover non-allocated but recoverable messages*, a mitigation has been discussed in academic literature in McGee (2022), which identifies that while non-allocated data in sms.db for Messages on iOS is often quickly overwritten, the "Biome directory" can also be reviewed for message information and content. This has resulted in the addition of a mitigation *M1077: Ensure potential secondary locations for stored message content are reviewed*, with McGee (2022) included in the references section of that mitigation, which is visible when reviewing the technique *T1072:Chat app examination* in the knowledge base.

5.2.2. File carving

As a second example, in *T1064:File carving*, one of the potential weaknesses identified is *W1106:Incorrect attribution of salvaged content to a current file system rather than a previous one*, which results in INAC-AS and MISINT. A recent paper applied digital stratigraphy to the challenge of attribution of carved content from recycled USB storage media (Schneider et al., 2024), so this has been added to the knowledge base as *M1061:Use digital stratigraphy to attempt to attribute data within a specific file system*, with a reference to the published work. This makes that work more visible and explicitly linked to a practical technique that would be used in an investigation.

5.3. Identifying potential weaknesses in an investigation

Extending the examples in the previous section, this example demonstrates the use of the knowledge base to reflect on the results of an entire digital investigation, or on a standardised process, and to highlight potential weaknesses or challenges to the process or evidence.

The script *generate_case_evaluation.py* in the repository can be supplied with the technique IDs used during an investigation, or that form part of a standardised process, and it will draw from the knowledge base, compile those techniques, together with the associated potential weaknesses, along with known mitigations. This allows a systematic review of the techniques used, consideration as to whether all potential weaknesses have been considered, and whether all *appropriate* known mitigations have been put in place during the investigation.

As an illustration, using a simple, synthetic, scenario-based disk image where it is necessary to determine if some 'illegal' content is present, Fig. 5 shows the use of the knowledge base to document at a high level the techniques that were used to analyse the digital evidence and reach conclusions (in a non-accredited lab environment without a tool testing programme).

A subset of the techniques used are highlighted in Fig. 5 in the appendix, and the full example is available in the repository. The techniques used during the investigation were: *T1042:Disk image hash validation*, *T1049:Keyword searching*, *T1054:Content review for relevant material*, *T1060:Process file system structures*, *T1061:Non-allocated file recovery*, *T1064:File carving*, *T1063:Identify file types*, *T1091:Bookmarking*, *T1092:Produce bookmark report*.

This example shows the basics of using the knowledge base to highlight where there may be unaddressed weaknesses in the investigation. In this case, there is a somewhat artificial problem with the case. If we consider the *T1042:Disk image hash validation*, it is possible to see that *W1128:Image replaced with tampered version along with updated stored hash* has been automatically flagged in the final column as having no mitigations in place. Since this sample image was downloaded from a repository of sample disk images with no information about any security mechanisms in place or audit trails (e.g. *M1022:Restrict access to stored images*), it is unsurprising that this is flagged as an unmitigated weaknesses. This is an unrealistic example, but illustrates how the main weaknesses can be automatically highlighted. A risk-based approach to addressing weaknesses is discussed in Section 7.2.

However, more realistically, even this simple example provides insights into the weaknesses in fairly standard digital forensic processes. Manual verification of results, tool testing, and multiple tool verification are often discussed as methods for ensuring correct results. This example highlights that these mitigations should not be considered interchangeable. For example, within *T1060:Process file system structures*, considering potential mitigations, *M1051:Validation of specific file system parsing* is likely to be extremely challenging to undertake comprehensively for smaller labs. *M1050:Manual verification of relevant data* is also an available mitigation, but considering a specific weakness e.g. *W1068: Failing to identify the existence of a live file*, it is difficult to imagine how manual verification could be comprehensively performed to check that no files were missed in reconstructing a large complex file system. This becomes further apparent in *T1049:Keyword searching*, where correctly adding and retrieving keyword from the index relies either on testing-based mitigations, or verification against live search options. In absence of tool testing programmes and manual verification being infeasible in some circumstances, multi-tool verification of results is the only remaining mitigation, but one that at present is very difficult due to inconsistent output from tools, highlighting the importance of standard representation initiatives such as the CASE Ontology, and its adoption in tools.

Recognising that such a case review requires additional, and often scarce resources, one time saving feature has been implemented which is to load a 'lab configuration'. If the examination is undertaken in an environment with Standard Operating Procedures (SOPs), tool testing, and validated processes, this can be captured in a configuration file which will auto-populate many of the fields, leaving only those which need consideration by the examiner for this specific case.

It is important to note that not all mitigations need to be implemented - a subset may be sufficient. In some cases there may not be any mitigations in place, and this is not a cause to reject evidence or results. It is simply a means of highlighting that one area of the investigation needs to be looked at closely and determine if the lack of mitigation or the presence of a *potential* weakness is significant to this case.

In Hargreaves (2009), as part of the 'accuracy' requirement for the reliability of digital evidence, it states "it should be possible to assess the amount of error associated with all techniques used to obtain and process digital evidence, and that amount of error should be acceptable in the context of the current investigation." The SOLVE-IT knowledge base can assist with the assessment of error associated with techniques, but determining whether it is acceptable in the context of an investigation is for practitioners and courts to decide.

5.4. CASE ontology interfacing

The knowledge base also provides an option for a new, technique-driven view on the CASE Ontology. Each technique has a property that can record the CASE Ontology entities that could be used to model selected output from the technique. For example: the technique *T1072: Chat app analysis*, could result in output that could be modelled by: *observable:Message*, *observable:SMSMessage*, and *observable:Contact*. The technique *T1005:Crime Scene Searching* potentially outputs: *observable: Device*, *observable:Computer*, *observable:MobileDevice* and other similar device types, and *T1035:Dictionary Attack* has the output *observable: password*.

Including the CASE entities improves visibility of the ontology and provides some additional indexing. It also highlights some gaps in the ontology when digital forensic techniques are considered in this systematic way. For example, in *T1069:Browser analysis*, the CASE classes of *observable:URLHistory*, *observable:URLHistoryEntry*, *observable:URLVisit*, *observable:CookieHistory*, *observable:BrowserCookie*, and *observable:*

BrowserBookmark can be referenced, but it has been observed that there is no specific CASE Ontology entry for browser cache entries.

5.5. Structured consideration of potential for AI

There have been several recent discussions of the use of AI in digital forensics from different perspectives, including a general overview e.g. Du et al. (2020), application of a specific technology such as ChatGPT (Scanlon et al., 2023), applications to specific areas such as network forensics (Rizvi et al., 2022) or report writing (Michelet and Breitinger, 2024). Recently Wickramasekara et al. (2024) considers specifically LLM applications and structured the discussion around a digital forensic process model to provide a systematic look at the possibilities. This provides inspiration to consider the applications of AI in a more structured way and to consider precisely where AI techniques are applicable or not, and what specific weaknesses exist when using that technique, so that they can be considered during any application of AI. This knowledge base at the granularity of 'digital forensic technique' allows this analysis, and focuses discussion at the specific technique, rather than a stage of a process model.

Fig. 6 in the appendix shows how the knowledge base can be used to consider available digital forensic techniques in a structured manner and consider the existing and potential ways in which AI could be used to assist. A summary is provided in Table 3. This review uses explicit categories (described below) rather then general ones e.g. high, medium or low. Also note that generic applications such as the use of AI to inform how to perform a technique or explain CLI tool usage are not included.

This systematic approach highlighted that of the 104 identified techniques, five have existing implementations in tools (e.g. *T1081: Image content analysis* available in Magnet AI (Magnet, 2018), seven have existing academic work with implementations (e.g. *T1083:Memory Examination (OS-level)* (Oh et al., 2024)), and 11 have existing academic work where an idea is proposed (e.g. *T1049:Keyword searching* (Scanlon et al., 2023). The status of the remaining techniques are either currently unclassified (n = 4), or it was deemed that a non-AI based process or a traditional programmatic approach is sufficient in 51 techniques e.g. *T1042:Disk image hash verification*. However, these latter classifications are subjective rather than reference/literature based. Nevertheless, this facilitates much more specific discussions around AI applications in digital forensics.

This may have gaps, and given the speed of developments in this area it may be out of date by the time the work is published, but it highlights that this knowledge base provides a detailed and structured approach to considering where a technology such as AI could be applied to digital forensics.

5.6. Academic research gaps

As a final example, the knowledge base can also be used as a means of situating research carried out, i.e. determining if a piece of research is an improvement to an existing technique (either identifying weaknesses and/or providing mitigations to those potential problems), or a new technique all together. However, more interestingly, from the techniques documented so far, several potential weaknesses have no documented/evidenced mitigations. These include:

- W1001: During a triage process, excluding a device that contains relevant information (INCOMP)
- W1015: Powering on SSD results in sectors being zeroed by TRIM operation (INCOMP, INAC-ALT, INAC-COR)
- W1038: Mobile backup process returns an incomplete set of backup data (INCOMP)

This could be that the information within the knowledge base is incomplete, in which case those researchers with relevant mitigations could update the details in the knowledge base to ensure that the mitigation is documented, or if a mitigation is not currently available, then this is clearly an important research area for the digital forensic community. In addition, some of the existing mitigations may not be optimal or work in all circumstances and further research may be able to improve upon the current mitigation options.

Table 3
Categorisation of each technique based on the applicability and current status of the use of AI for that purpose.

Classification	Count (n = 104)
Unclassified	**4**
Non AI-based process likely sufficient	**51**
Some application can be envisaged	**26**
In academic work (idea)	**11**
In academic work (implementation)	**7**
In tools	**5**

6. Crowdsourcing techniques, weaknesses and mitigations

The SOLVE-IT knowledge base is hosted on Github, which allows crowdsourced contributions, either identifying new techniques or improving the details of existing ones. This can be achieved through pull requests, or allows 'Issues' to be reported via the Github interface providing a less "hands on" approach to updating the model.

To test the feasibility of this work as a crowdsourced project, several researchers who were not involved in the development of the knowledge base were personally contacted and asked to contribute to techniques which they were very familiar with. These additions to the knowledge base are shown in Table 4, and are included in the acknowledgements, demonstrating that community effort towards this initiative is feasible.

7. Discussion

7.1. Limitations

This work presents an ambitious starting point for a valuable and versatile resource. As a 'starting point' there are practical limitations of scope, scale, and detail. There are mitigations, weaknesses, techniques, and likely even categories of techniques that are not currently included. However, as has already been stressed, this is not intended as a complete knowledge base, rather an argument that such a resource has immense value and versatility, and that existing knowledge bases such as ATT&CK can be used as inspiration for such a resource. The limited scope and detail is by design and by necessity and highlights the need for this to be a crowdsourced project.

In terms of the implementation of the knowledge base there are other limitations. Several fields are free text, and while techniques, weaknesses, and mitigations are broken out as separate entities, there may be benefit to indexing other pieces of data too, e.g. references, examples,

Table 4
A list of contributions to the knowledge base made by third-party researchers, who are included in the acknowledgements.

Technique	Community Addition Fields
Memory imaging (T1003)	errors, mitigations, references (Case and Richard III, 2017; Latzo et al., 2019; Pagani et al., 2019; Ottmann et al., 2023b,a; Rzepka et al., 2024; Campbell, 2013; Vömel and Stüttgen, 2013; Gruhn and Freiling, 2016; Martignoni et al., 2010)
Brute force attack (T1034)	definition, errors, mitigations, references (Kanta et al., 2021b)
Dictionary attack (T1035)	definition, errors, mitigations, references (Kanta et al., 2021a, 2023)
Timeline generation (T1052)	errors, mitigations, references (Dreier et al., 2024; Vanini et al., 2024)
Timeline analysis (T1086)	errors, mitigations, references (Sandvik and Årnes, 2018; Bollé et al., 2020; Vanini et al., 2024)

tools, and datasets. This would improve searchability and filtering. Extrapolating tools as distinct entities, and providing a field for SOPs would also facilitate mapping into the methods, tools and procedures concepts in the *Forensic Field Map* in van Beek (2018). It may also be that other fields are necessary e.g. the CASE output entities of a technique are currently documented, but no use for a CASE input class has been identified so far as many are simply 'a file'. However, there is scope to update the knowledge base schema for any field for which added value can be demonstrated.

The crowdsourcing implementation also needs further consideration. The precise mechanism for ensuring quality in the knowledge base is still under consideration until it is established which, if any, organisation the knowledge-base would be hosted within. There are numerous mechanisms available, some of which can be derived from Casey et al. (2022) e.g. voting, or taking inspiration from CASE, having a Technical Steering Committee, or the practicalities of issue tracking in the Unified Cyber Ontology (UCO, 2024) along with a 'Solutions Approval' process. In addition, further documented guidance is needed on the style and detail expected for techniques, weaknesses, and mitigations.

It is also unclear as to the granularity needed within SOLVE-IT e.g. whether *T1072:Chat App Examination* should extend into subtechniques representing the analysis of individual chat apps. At present the granularity of the knowledge base has remained at a relatively high abstraction layer, but documenting specifics may be advantageous. Fortunately, the sub-technique implementation facilitates this if desired. It may also be that different versions of the matrix are needed for different areas of digital forensics as the knowledge base expands.

In terms of the demonstrations included in the paper, error-focused datasets are not a new concept (e.g. the Digital Forensics Tool Testing images (Carrier, 2010)), but the renewed effort to create public versions is relatively recent. e.g. Hargreaves et al. (2024b). This paper has indexed the dataset from that paper, and has identified some others that need to be developed, but has not created any new ones. There is much work to do in this area, but this was not the specific focus of the paper. However, it has provided a structure in which datasets that capture a weakness can be indexed and applied to tool testing programmes.

The current indexing of weaknesses in the knowledge base considers the nature of the resulting problem (from ASTM (2018) e.g. Incompleteness, Inaccuracy:Existence, Inaccuracy:Association, Inaccuracy: Corruption, Inaccuracy:Alteration, and Misinterpretation). However, another approach is to consider the sources of uncertainty (Ryser, 2024): e.g. environment, data, methodology, knowledge, semantic, expert, probabilities, tools, forensic process. Indexing weaknesses based on these classifications could be added, or perhaps used as a framework to better identify weaknesses that are then indexed based on their resulting problem.

Regarding the evaluation of a case against the weaknesses in techniques used, this is recognised as a substantial overhead to an investigation. However, this work presents this as tool that can be used to pre-emptively identify problems and could in some cases even result in saved time later in the investigation if challenges are pre-empted. In addition, some automation has been provided to reduce the overhead in labs with SOPs and tool testing programmes. Alternatively, reviews of SOPs themselves could potentially be conducted using this approach rather than specific investigations, which would be more efficient.

7.2. Future work

In terms of further work, the obvious effort is to improve the structure of the knowledge base and continue to populate it. In addition, there are also other potential uses of the knowledge base that have not yet been demonstrated.

- With an index of weaknesses, it may be possible to conduct risk analysis to consider the likelihood and impact of individual weaknesses to determine where to best apply resources.
- In teaching, this breakdown of techniques and the ordering by objective may help illustrate to students the options available during an investigation. Stressing the weaknesses of particular techniques at this point may emphasise critical thinking about the processes they are applying.
- Lab capability assessments - a lab could consider the techniques available and consider if the current set of tools in use provides full coverage of the techniques that can be performed. This could also help create uniform documentation for techniques in use.
- Skills assessments - SOLVE-IT could be used to form the basis for a structured knowledge and skills assessment for new examiners.

Other extensions are possible since the high-level categories in SOLVE-IT describe objectives within a digital investigation, however, they do not represent the highest level goals that an investigator may have such as "Determine how a file got onto this machine" or "who sent this email". With all the techniques in place and documented, pathways through several techniques to answering these questions could be developed. Finally, in terms of expansion, scalability, and resilience, if the knowledge base becomes a central, useful digital forensic resource, more management will become necessary. In this case, perhaps adoption by an organisation similar to MITRE, NIST, or adoption by CASE or DFRWS may be advantageous to provide the additional management effort required.

8. Conclusions

This paper has shown that a digital forensic knowledge base inspired by MITRE ATT&CK is feasible and useful for mitigating weaknesses in lab processes, individual investigations, evidence interpretation, tool testing, education, and professional training. This has been demonstrated through a prototype design and implementation, and the paper has also shown several use cases where such a knowledge base would be of value to the community. Digital forensic practitioners and labs can use SOLVE-IT to mitigate weaknesses during their work, either through review of specific complex cases, or review of standard processes. It has also evidenced that it can be populated as a crowdsourced project through contributions from several researchers not involved in creating the knowledge base.

MITRE ATT&CK has become a valuable and versatile resource within the cybersecurity communities, and the digital forensics community would greatly benefit from a similar resource. This paper provides the starting point.

Acknowledgements

Thank you to Jessica Hyde for suggesting the iOS Messages example, Alex Nelson for assistance with the CASE examples, Gill Tully for helpful discussions about risk analysis and applications of the knowledge base to quality assurance. Also many thanks to Jenny Ottmann, Katerina Kanta, and Céline Vanini for their community additions to the knowledge base.

Appendix

Technique name:	Disk imaging	back to main						
Technique ID:	T1002							
Category:	['Acquire']							
Description:	Copying of sectors from a storage media, typically LBA0 to LBAmax into an imaging format. The could be from a traditional hard disk, SSD, USB stick, or data from an eMMC chip that has been desoldered and placed in a reader.							
Synonyms:	[]							
Details:								
Subtechniques:	[]							
CASE output entities:	['observable:Image']							
Examples:	['dcfldd', 'FTK Imager', 'Magnet ACQUIRE']							
Potential weaknesses:								
Weakness ID:	**Detail:**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**MISINT**	**Mitigations**
W1004	Acquisition does not include all sectors from LBA0 to LBA max	x						M1003,
W1006	Acquistion does not include data in HPA	x						M1005,
W1007	Acquistion does not include data in DCO	x						M1006,
W1013	Acquisition includes extra bytes		x					M1003,M1009,
W1014	Imaging process changes original data				x			M1007,M1008,
W1015	Powering on SSD results in sectors being wiped by TRIM operation	x			x	x		
W1016	Data copied from sectors on source are stored incorrectly				x	x		M1009,
W1136	Not recovering data from a failed hard drive	x						M1089,
W1143	Acquisition method does not read remapped sectors e.g. G-Lists	x						M1102,
Mitigations:								
M1003	Check image size corresponds with drive label							
M1005	Testing to ensure software and hardware setup detects HPAs							
M1006	Testing to ensure software and hardware setup detects DCOs							
M1007	Use hardware write blocker (T1012)							
M1008	Use software write blocker (T1013)							
M1009	Check hash of image matches hash of source material							
M1089	Attempt physical disk repair							
M1102	Apply techqniues to read remapped sectors							
References:	Nikkel, B., 2016. Practical forensic imaging: securing digital evidence with Linux tools. No Starch Press, Chapter 6, "Forensic Image Acquisition							['T1002']
	Gupta, M.R., Hoeschele, M.D. and Rogers, M.K., 2006. Hidden disk areas: HPA and DCO. International Journal of Digital Evidence, 5(1), pp.1-8.							['W1006', 'W1007']
	DSTL Digital Forensics Bulletin, TRIM & Garbage Collection on SSDs, Edition 27, July 2024 DSTL/PUB160897, https://mailchi.mp/3b5bda81ff5f/digital-forensic-bulletin-edition27-trim-and-garbage-collection-on-ssds9445401							['W1015']
	Turner, P. The Remapped/Reallocated Sector Conundrum, Presentation at DFRWS US 2024, https://www.dfrws.org/wp-content/uploads/2023/07/turner-remappedsectors.pdf							['W1143', 'M1102']

Fig. 3. An illustrative example of the detail within a specific technique. This shows the disk imaging technique within the acquisition objective. Nine potential weaknesses have been identified so far, along with eight indexed mitigations. References that are indexed are associated with either the technique in general, specific weaknesses, or specific mitigations (DSTL, 2024; Gupta et al., 2006; Nikkel, 2016; Turner, 2024). Some mitigations link to techniques within the knowledge base e.g. *M1007: Use hardware write blocker*.

Technique name:	Content review for relevant material	back to main						
Technique ID:	T1054							
Category:	['Locate Relevant Digital Artefacts']							
Description:								
Synonyms:	[]							
Details:								
Subtechniques:	['T1079', 'T1080', 'T1081', 'T1082']							
CASE output entities:	[]							
Examples:	[]							
Potential weaknesses:								
Weakness ID:	**Detail:**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**MISINT**	**Mitigations**
W1060	A relevant piece of media was not flagged as relevant	X		X				M1039,M1040,M1041,
W1061	An irrelevant piece of media was flagged as relevant			X				M1039,M1040,M1041,M1042,
W1062	The grade or specific tag given to item is incorrect			X				M1039,M1040,M1041,M1042,
Mitigations:								
M1039	Hash matching to reduce need for manual review							
M1040	Use of AI to flag potentially relevant content to reduce need for manual review							
M1041	Regular breaks to minimise investigator fatigue							
M1042	Secondary review of flagged items to ensure relevance							

Fig. 4. Illustrates the media review technique within the 'Locating Relevant Digital Artefacts. Here non technical mitigations can also be captured e.g. *M1041: Regular breaks to minimise investigator/reviewer fatigue.*'

	Potential Weaknesses	INCOMP	INAC-EX	INAC-AS	INAC-ALT	INAC-COR	MISINT	Mitigations								Y	N	-	NA	Met	Status
T1042: Disk image hash verification	**Potential Weaknesses**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**INAC-MISINT**	**M1021 Verify the disk image integrity with multiple hash algorithms e.g. MD5 and SHA1 (Kessler 2016)**	**M1022 Restrict access to stored disk images**	**M1023 Ensure and check logs of access to stored disk images**	**M1070 Ensure hash algorithm(s) used are resistant to collisions through data manipulation**	**M1075 Testing programme to validate hashes of data in images is calculated correctly**	**M1085 Use of multiple tools to verify disk image hash**	**M1076 Testing programme to validate hashes of metadata in images is calculated correctly**	**M1074 Validate image hash against one stored externally to the image in a trusted location.**						
W1042	Disk image was tampered with, but manipulated to have a collision with original hash					x		N	N	N	N					0	4	0	0	0/4	x
W1124	Failure to compute hash correctly: this could result in a message indicating corrupt evidence, thus stopping or delaying further investigation			X								N	Y			1	1	0	0	1/2	
W1125	Failure to validate hash properly: this could allow errors from earlier to propagate e.g. incorrect sectors					X						N	Y			1	1	0	0	1/2	
W1126	Failure to validate hash properly allowing a an incomplete disk image to present as validated	X										N	Y			1	1	0	0	1/2	
W1127	Failure to validate metadata: this could allow details such as acquisition date to be changed					X								NA		0	0	0	1	0/0	
W1128	Image replaced with tampered version along with updated stored hash					X			N	N					N	0	3	0	0	0/3	x
T1060: Process file system structures	**Potential Weaknesses**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**INAC-MISINT**	**M1050 Manual verification of relevant data**	**M1051 Validation of specific file system parsing operation for the file system under consideration**	**M1052 Use of a second independent tool to compare live file system listing**	**M1053 Verification of the file system specification used as the basis for the tool development**										
W1068	Failing to identify the existence of a live file	X						N	N	Y	N					1	3	0	0	1/4	
W1069	Failing to identify the existence of a live directory	X						N	N	Y	N					1	3	0	0	1/4	
W1070	Failing to recover the complete contents of a live file	X						NA	NA	NA	NA					0	0	0	4	0/0	
W1071	Failing to recover a complete listing of a live directory	X						N	N	Y	N					1	3	0	0	1/4	
W1072	Recovering incorrect contents of a live file				X			NA	NA	NA	NA					0	0	0	4	0/0	
W1073	Recovering an incorrect listing of a live directory				X			N	N	Y	N					1	3	0	0	1/4	
W1074	Attributing a file to the wrong directory			X				N	N	Y	N					1	3	0	0	1/4	
W1075	Presenting a file that does not exist		X					N	N	Y	N					1	3	0	0	1/4	
W1076	Presenting a directory that does not exist		X					N	N	Y	N					1	3	0	0	1/4	
W1077	Presenting a deleted file as a live file			X			X	NA	NA	NA	NA					0	0	0	4	0/0	
W1078	Presenting a live file as deleted			X			X	N	N	Y	N					1	3	0	0	1/4	
W1079	Recovering incorrect metadata about a file				X			Y	N	Y	N					2	2	0	0	2/4	
W1080	Recovering incorrect metadata about a directory				X			N	N	Y	N					1	3	0	0	1/4	
W1081	Failing to recover file system metadata	X						Y	N	Y	N					2	2	0	0	2/4	
W1082	Recovering file system metadata incorrectly				X			Y	N	Y	N					2	2	0	0	2/4	
W1083	Failure to recover additional specialised file system specific content e.g. Alternate Data Streams	X						NA	NA	NA	NA					0	0	0	4	0/0	
W1084	Failure to correctly recover specialised file system specific content e.g. Alternate Data Streams				X			NA	NA	NA	NA					0	0	0	4	0/0	
T1061: Non-allocated file recovery	**Potential Weaknesses**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**INAC-MISINT**	**M1084 Test that the tool used expresses uncertainty in the classification of file recovery results**	**M1079 Analyse file content to detect incompatibility between the content and the recovered file system metadata**	**M1080 Use a hashset of known files to detect that recovered content does not match the expected file content**	**M1081 Check if recovered content matches with size in metadata e.g. file footer at expected location.**	**M1082 Analyse file system to find any duplicate references to blocks that are referenced by the non-allocated file**	**M1083 Analyse content for embedded timestamps to find that content is a newer file than indicated by metadata (beware of file tunneling)**							
W1133	Tool fails to express uncertainty in the classification of file recovery results, e.g., where content may be partly overwritten.						X	Y								1	0	0	0	1/1	
W1134	File metadata and name of a non-allocated file are recovered, but the content is fragmented/overwritten, but is still presented as the content of the recovered file			X			X		Y	N	Y					2	1	0	0	2/3	
W1135	File metadata and name of a non-allocated file are recovered, but its content has been overwritten by a newer file, but the non-allocated file's content is presented as this newer file's data			X			X		Y	N	Y	Y	N			3	2	0	0	3/5	
T1064: File carving	**Potential Weaknesses**	**INCOMP**	**INAC-EX**	**INAC-AS**	**INAC-ALT**	**INAC-COR**	**INAC-MISINT**	**M1078 Use of multiple carving tools and comparing results**	**M1062 Verify semantic integrity of carved content using manual inspection of relevant data**	**M1063 Verify semantic integrity of carved content using an automated tool**	**M1064 Verify that salvaged content is renderable using visual inspection of displayed/rendered content**	**M1090 Compare carved results against known files (e.g. from another source)**	**M1061 Use digital stratigraphy to attempt to attribute data within a specific file system**	**M1060 Repair salvaged content by grafting different reference data onto fragment(s)**	**M1065 Render salvaged content by reconstructing a container around fragment(s) ...**						
W1103	Failure to carve salvageable content	X						Y								1	0	0	0	1/1	
W1104	Incorrect carving of a complete file				X	X			N	N	Y	NA				1	2	0	1	1/3	
W1105	Incorrect reassembly of a file				X	X			N	N	Y	NA				1	2	0	1	1/3	
W1106	Incorrect attribution of salvaged content to a current file system rather than a previous one			X			X						N			0	1	0	0	0/1	x
W1107	Failure to repair a content fragment	X				X								NA	NA	0	0	0	2	0/0	

Fig. 5. An extract from an example review of an investigation using the SOLVE-IT knowledge base and the *generate_case_review.py script.*

ID	Technique Name	AI Status	Example (where applicable)
T1001	Triage	Ac-idea	Identifying most relevant devices could potentially be improved with AI (Du et al 2020)
T1002	Disk Imaging	NonAI	
T1003	Memory Imaging	AppEnv	Perhaps non-linear imaging could minimise smearing and AI may help identify locations to prioritise
T1004	Selective data acquistion	AppEnv	Use of AI to scan for relevant content, determining content prioritised for acquisition
T1005	Crime Scene Searching	Ac-idea	Help identifying pieces of evidence e.g. with VisionLLM (Wickramasekara et al 2024)
T1006	Digital Sniffer Dogs	Unclassified	
T1007	SyncTriage	AppEnv	Perhaps AI could match references to other devices in addition to deterministic matching
T1008	Profiling Network Traffic	Ac-imp	Identifying potentially infected machines from network traffic(Gratian et al 2019)
T1009	Locate Cloud Account Identifiers	AppEnv	Perhaps AI could match cloud identifiers in addition to deterministic matching
T1010	Place device in faraday environment	NonAI	
T1011	Evidence bags	NonAI	
T1012	Hardware write blockers	NonAI	
T1013	Software write blockers	NonAI	
T1014	Chain of custody documentation	AppEnv	Perhaps automatically highlighting inconsistences or errors in documentation
T1015	Privacy preserving selective extraction	Ac-Idea	Perhaps AI could match relevant content without human review to provide some privacy protections (Webb et al 2024)
T1016	Live data collection	Unclassified	
T1017	Network packet capture	NonAI	
T1018	Remote data collection	NonAI	
T1019	Mobile backup extraction	NonAI	
T1020	Mobile file system extraction	NonAI	
T1021	Mobile storage imaging	NonAI	
T1022	Mobile device screenshot based capture	In tools	OCR already can be used to extract text from images from screenshot based capture (e.g X-Ways)
T1023	Cloud data collection using account details	NonAI	
T1024	Cloud data collection via request	NonAI	
T1025	Writing data to a forensic image format	NonAI	
T1026	Writing data in standard archive format	NonAI	
T1027	Data read using JTAG	AppEnv	Perhaps AI assistance with finding JTAG TAPs
T1028	Chip-off	AppEnv	Perhaps analysing data from thermal experiments to minimise chip damage
T1029	Data read from desoldered eMMC	NonAI	
T1030	Data read from unmanaged NAND	NonAI	
T1031	Key recovery from memory	NonAI	
T1032	Side channel	Ac-imp	Identifying activites on IoT devices using ML-based side channel attacks(Le et al 2021)
T1033	Extraction of account details from an accessible device	NonAI	
T1034	Brute force attack	NonAI	
T1035	Dictionary attack	Ac-imp	GAN for password generation (Hitaj et al 2019)
T1036	Smudge attack	AppEnv	Perhaps image processing to infer lock code or pattern
T1037	Obtain password from suspect	Unclassified	
T1038	Rainbow tables	NonAI	
T1039	App Downgrade	NonAI	
T1040	Use mobile device exploit	NonAI	
T1041	Pin2Pwn	AppEnv	Perhaps AI assistance analysing PIN out
T1042	Disk image hash verification	NonAI	
T1043	Forensic image format decoding	NonAI	
T1044	Mobile backup decoding	NonAI	
T1045	Decode standard archive format	NonAI	
T1046	Privileged material protection	AppEnv	Perhaps searching for and matching privileged material
T1047	Hash matching (reduce)	NonAI	
T1048	Privacy protection via partial processing	AppEnv	Perhaps in determining which aspects should be partially processed
T1049	Keyword searching	Ac-idea	Generating keyword lists (Scanlon et al 2023)
T1050	Hash matching (locate)	NonAI	
T1051	Fuzzy hash matching	NonAI	
T1052	Timeline generation	Ac-idea	Clock anomaly detection, (Du et al 2020)
T1053	Entity Extraction	AppEnv	Identifying names, addresses and other entities with AI rather than pattern matching
T1054	Content review for relevant material	Ac-idea	Summarisation, and identifying specific content types (Scanlon et al 2023)
T1055	File system content inspection	Ac-idea	Summarisation, and identifying specific content types (Scanlon et al 2023)
T1056	Entity connection identification	NonAI	
T1057	Steganography detection	Ac-idea	
T1058	Mismatched file extension detection	NonAI	
T1059	Identify partitions	NonAI	
T1060	Process file system structures	NonAI	
T1061	Non-allocated file recovery	NonAI	
T1062	Decryption of encrypted file systems/volumes	NonAI	
T1063	Identify file types	NonAI	
T1064	File carving	Ac-imp	e.g. file fragment identification (Alam & Demir 2024)
T1065	Content indexer examination (OS)	NonAI	
T1066	Log file examination (OS)	Ac-idea	Identifying anomalies (Scanlon et al 2023)
T1067	Cloud synchronisation feature examination (OS)	NonAI	
T1068	Recently used file identification (OS)	NonAI	
T1069	Browser examination	AppEnv	Reviewing browser history and identifying items of interest or summarisation
T1070	Email examination	AppEnv	Reviewing emails and identifying items of interest or summarisation
T1071	Database examination	AppEnv	Identifying critical tables in the schema
T1072	Chat app examination	In tools	Identification of grooming (e.g. Magnet AI)
T1073	Calendar app examination	AppEnv	Reviewing calendar entries and identifying items of interest or summarisation
T1074	Social network app examination	AppEnv	Inferring nature of the social network
T1075	Mapping app examination	NonAI	
T1077	Photos app examination	In tools	Finding images with specific type of content (e.g. Magnet AI)
T1078	Cloud sync app examination	NonAI	
T1079	Audio contnet analysis	AppEnv	Perhaps speaker identification, or detection of AI generated speech
T1080	Video content analysis	In tools	Deep fake identification (e.g. Amped Authenticate - Amped Software)
T1081	Image content analysis	In tools	Finding images with specific type of content (e.g. Magnet AI)
T1082	Document content analysis	AppEnv	Reviewing document content and identifying items of interest or summarisation
T1083	Memory examination (OS level)	Ac-imp	Use of LLMS to detect identify ransomware-related processes within memory dumps (Oh et al 2024)
T1084	Extraction of user accounts	NonAI	
T1085	Identify conflation	NonAI	
T1086	Timeline analysis	Ac-imp	Anomaly detection in forensic timelines (Studiawan & Sohel 2021)
T1087	Geospatial analysis	AppEnv	Assistance in finding patterns in location information
T1088	Connection analysis	Ac-idea	Graph Neural Networks (GNN) to model interesting relation graphs (Hensler & Hyde 2019)
T1089	Source code review	Ac-idea	Explaining source code operations (Chen at al 2023)
T1090	Experimentation	AppEnv	Identifying relevant traces from a set of actions carried out in an experiment
T1091	Bookmarking	AppEnv	Perhaps some automated bookmarking based on investigator specified criteria
T1092	Produce bookmark-based automated report	NonAI	
T1093	Write expert report	Ac-imp	LLMs (Llama-2 & ChatGPT-3.5) to genereate reports from tool output (Michelet & Breitinger 2024)
T1094	Disclosure	NonAI	
T1095	Instrumentation	NonAI	
T1096	Run programs identification (OS)	NonAI	
T1097	Installed programs identification (OS)	AppEnv	Perhaps separating default programs from installed ones or identifying ones of interest
T1098	User account analysis (OS)	NonAI	
T1099	File Repair with Grafting	Unclassified	
T1100	EXIF data analysis	NonAI	
T1101	Cell Site Survey	NonAI	
T1102	Decode data from image from unmanaged NAND	AppEnv	Perhaps finding patterns in unknown NAND types
T1103	Virtualise suspect system for previewing	NonAI	
T1104	Collect data using open source intelligence	AppEnv	Suggesting other related places where information may be available based on content reviewed so far
T1105	Memory examination (application level)	AppEnv	Identifying relevant memory structures within application memory space

Fig. 6. Demonstration of the use of the knowledge base to provide more granular review of the status of AI applications in digital forensics. This is not intended to be a complete systematic review and only one example is provided for each technique to evidence the highest level of implementation identified (Alam and Demir, 2024; Chen et al., 2023; Du et al., 2020; Gratian et al., 2019; Henseler and Hyde, 2019; Hitaj et al., 2019; Le et al., 2021; Michaylov, 2023; Michelet and Breitinger, 2024; Oh et al., 2024; Scanlon et al., 2023; Studiawan and Sohel, 2021; Webb et al., 2024; Wickramasekara et al., 2024).

References

Ahmed, M., Panda, S., Xenakis, C., Panaousis, E., 2022. Mitre att&ck-driven cyber risk assessment. In: Proceedings of the 17th International Conference on Availability, Reliability and Security, pp. 1–10.

Al-Dhaqm, A., Ikuesan, R.A., Kebande, V.R., Abd Razak, S., Grispos, G., Choo, K.K.R., Al-Rimy, B.A.S., Alsewari, A.A., 2021. Digital forensics subdomains: the state of the art and future directions. IEEE Access 9.

Alam, S., Demir, A.K., 2024. Sift: sifting file types—application of explainable artificial intelligence in cyber forensics. Cybersecurity 7, 52.

ASTM, 2018. Astm e3016-18 standard guide for establishing confidence in digital and multimedia evidence forensic results by error mitigation analysis. https://www.astm.org/e3016-18.html.

Barnhart, H., 2017. Time is not on our side when it comes to messages in ios 11. https://smarterforensics.com/2017/09/time-is-not-on-our-side-when-it-comes-to-messages-in-ios-11/.

Beebe, N.L., Clark, J.G., 2005. A hierarchical, objectives-based framework for the digital investigations process. Digit. Invest. 2, 147–167.

BinaryHick, 2022. New msgstore – who 'dis? a look at an updated whatsapp on android. https://thebinaryhick.blog/2022/06/09/new-msgstore-who-dis-a-look-at-an-updated-whatsapp-on-android/.

Bollé, T., Casey, E., Jacquet, M., 2020. The role of evaluations in reaching decisions using automated systems supporting forensic analysis. Forensic Sci. Int.: Digit. Invest. 34, 301016.

Breitinger, F., Hilgert, J.N., Hargreaves, C., Sheppard, J., Overdorf, R., Scanlon, M., 2024. Dfrws eu 10-year review and future directions in digital forensic research. Forensic Sci. Int.: Digit. Invest. 48.

Campbell, W., 2013. Volatile memory acquisition tools–a comparison across taint and correctness. In: Proceedings of the 11th Australian Digital Forensics Conference. ADF 2013.

Carrier, B., 2010. Digital forensics tool testing images. https://dftt.sourceforge.net.

Carrier, B., et al., 2003. Defining digital forensic examination and analysis tools using abstraction layers. Int. J. Digit. Eviden. 1, 1–12.

CASE, 2024. Cyber-investigation analysis standard expression (case). https://caseontology.org.

Case, A., Richard III, G.G., 2017. Memory forensics: the path forward. Digit. Invest. 20, 23–33.

CASE Ontology, 2024a. action:actionlifecycle. https://ontology.caseontology.org/documentation/class-actionactionlifecycle.html.

CASE Ontology, 2024b. Case forensic lifecycle example. https://github.com/casework/CASE-Examples/blob/master/examples/illustrations/forensic_lifecycle/forensic_lifecycle.json.

CASE Ontology, 2024c. investigation:investigativeaction. https://ontology.caseontology.org/documentation/class-investigationinvestigativeaction.html.

CASE Ontology, 2024d. observable:image. https://ontology.caseontology.org/documentation/class-observableimage.html.

Casey, E., Barnum, S., Griffith, R., Snyder, J., van Beek, H., Nelson, A., 2017. Advancing coordinated cyber-investigations and tool interoperability using a community developed specification language. Digit. Invest. 22.

Casey, E., Nguyen, L., Mates, J., Lalliss, S., 2022. Crowdsourcing forensics: creating a curated catalog of digital forensic artifacts. J. Forensic Sci. 67, 1846–1857.

Chen, E., Huang, R., Chen, H.S., Tseng, Y.H., Li, L.Y., 2023. Gptutor: a chatgpt-powered programming tool for code explanation. In: International Conference on Artificial Intelligence in Education. Springer, pp. 321–327.

Collie, J., 2018. Digital forensic evidence-flaws in the criminal justice system. Forensic Sci. Int. 289, 154–155.

Cusack, B., Homewood, A., 2013. Identifying bugs in digital forensic tools. In: Proceedings of the 11th Australian Digital Forensics Conference.

Dreier, L.M., Vanini, C., Hargreaves, C.J., Breitinger, F., Freiling, F., 2024. Beyond timestamps: integrating implicit timing information into digital forensic timelines. Forensic Sci. Int.: Digit. Invest. 49.

DSTL, 2024. Dstl Digital Forensics Bulletin, Trim & Garbage Collection on Ssds, 27. july 2024 dstl/pub160897. https://mailchi.mp/3b5bda81ff5f/digital-forensic-bulletin-edition27-trim-and-garbage-collection-on-ssds9445401.

Du, X., Hargreaves, C., Sheppard, J., Anda, F., Sayakkara, A., Le-Khac, N.A., Scanlon, M., 2020. Sok: exploring the state of the art and the future potential of artificial intelligence in digital forensic investigation. In: Proceedings of the 15th International Conference on Availability, Reliability and Security, pp. 1–10.

Gratian, M., Bhansali, D., Cukier, M., Dykstra, J., 2019. Identifying infected users via network traffic. Comput. Secur. 80, 306–316.

Gruhn, M., Freiling, F.C., 2016. Evaluating atomicity, and integrity of correct memory acquisition methods. Digit. Invest. 16, S1–S10.

Gupta, M.R., Hoeschele, M.D., Rogers, M.K., 2006. Hidden disk areas: hpa and dco. Int. J. Digit. Eviden. 5, 1–8.

Hargreaves, C., 2009. Assessing the Reliability of Digital Evidence from Live Investigations Involving Encryption. Ph.D. thesis. Cranfield University, UK.

Hargreaves, C., Breitinger, F., Dowthwaite, L., Webb, H., Scanlon, M., 2024a. Dfpulse: the 2024 digital forensic practitioner survey. Forensic Sci. Int.: Digit. Invest. 51.

Hargreaves, C., Nelson, A., Casey, E., 2024b. An abstract model for digital forensic analysis tools-a foundation for systematic error mitigation analysis. Forensic Sci. Int.: Digit. Invest. 48, 301679.

Henseler, H., Hyde, J., 2019. Technology assisted analysis of timeline and connections in digital forensic investigations. In: LegalAIIA@ ICAIL.

Hitaj, B., Gasti, P., Ateniese, G., Perez-Cruz, F., 2019. Passgan: a deep learning approach for password guessing. In: Applied Cryptography and Network Security: 17th International Conference, ACNS 2019, Bogota, Colombia, June 5–7, 2019, Proceedings 17. Springer, pp. 217–237.

Horsman, G., 2024. Sources of error in digital forensics. Forensic Sci. Int.: Digit. Invest. 48, 301693.

Horsman, G., Sunde, N., 2022. Unboxing the digital forensic investigation process. Sci. Justice 62, 171–180.

ISO/IEC, 2015. Iso/iec 27042:2015 Guidelines for the Analysis and Interpretation of Digital Evidence.

Kanta, A., Coisel, I., Scanlon, M., 2021a. Pcwq: a framework for evaluating password cracking wordlist quality. In: International Conference on Digital Forensics and Cyber Crime. Springer, pp. 159–175.

Kanta, A., Coisel, I., Scanlon, M., 2023. Harder, better, faster, stronger: optimising the performance of context-based password cracking dictionaries. Forensic Sci. Int.: Digit. Invest. 44, 301507.

Kanta, A., Coray, S., Coisel, I., Scanlon, M., 2021b. How viable is password cracking in digital forensic investigation? analyzing the guessability of over 3.9 billion real-world accounts. Forensic Sci. Int.: Digit. Invest. 37, 301186.

Karie, N.M., Venter, H.S., 2014. Toward a general ontology for digital forensic disciplines. J. Forensic Sci. 59, 1231–1241.

Latzo, T., Palutke, R., Freiling, F., 2019. A universal taxonomy and survey of forensic memory acquisition techniques. Digit. Invest. 28, 56–69.

Le, Q., Miralles-Pechuán, L., Sayakkara, A., Le-Khac, N.A., Scanlon, M., 2021. Identifying internet of things software activities using deep learning-based electromagnetic side-channel analysis. Forensic Sci. Int.: Digit. Invest. 39, 301308.

Magnet, 2018. Magnet.ai. https://www.magnetforensics.com/resources/magnet-axiom-2-0-magnet-ai/.

Martignoni, L., Fattori, A., Paleari, R., Cavallaro, L., 2010. Live and trustworthy forensic analysis of commodity production systems. In: International Workshop on Recent Advances in Intrusion Detection. Springer.

McGee, J., 2022. An alternate location for deleted SMS/iMessage data in apple devices. DFIR Review. https://dfir.pubpub.org/pub/yp6efc8q.

Michaylov, K., 2023. Exploring the Use of Steganography and Steganalysis in Forensic Investigations for Analysing Digital Evidence. B.S. thesis. University of Twente.

Michelet, G., Breitinger, F., 2024. Chatgpt, llama, can you write my report? an experiment on assisted digital forensics reports written using (local) large language models. Forensic Sci. Int.: Digit. Invest. 48.

MITRE, 2024a. Mitre att&ck enterprise matrix. https://attack.mitre.org/matrices/enterprise/.

MITRE, 2024b. Mitre att&ck key concepts. https://attack.mitre.org/resources/.

MITRE, 2024c. Mitre d3fend matrix. https://d3fend.mitre.org.

Nikkel, B., 2016. Practical Forensic Imaging: Securing Digital Evidence with Linux Tools. No Starch Press.

NIST, 2022. Nist internal report 8354: digital investigaiton techniques: a nist scientific foundation review. https://nvlpubs.nist.gov/nistpubs/ir/2022/NIST.IR.8354.pdf.

Oh, D.B., Kim, D., Kim, H.K., 2024. volgpt: evaluation on triaging ransomware process in memory forensics with large language model. Forensic Sci. Int.: Digit. Invest. 49, 301756.

Ottmann, J., Breitinger, F., Freiling, F., 2023a. An experimental assessment of inconsistencies in memory forensics. ACM Transact. Privac. Sec. 27, 1–29.

Ottmann, J., Cengiz, Ü., Breitinger, F., Freiling, F., 2023b. As if time had stopped–checking memory dumps for quasi-instantaneous consistency. arXiv preprint arXiv:2307.12060.

Pagani, F., Fedorov, O., Balzarotti, D., 2019. Introducing the temporal dimension to memory forensics. ACM Transact. Privac. Securit. (TOPS) 22, 1–21.

Palmer, G., et al., 2001. A road map for digital forensic research. In: First Digital Forensic Research Workshop, Utica, pp. 27–30 new york.

Rizvi, S., Scanlon, M., Mcgibney, J., Sheppard, J., 2022. Application of artificial intelligence to network forensics: survey, challenges and future directions. IEEE Access 10, 110362–110384.

Ryser, E., 2024. Facteurs d'incertitude en science forensique numérique (Doctoral dissertation). Phd thesis. Université de Lausanne. Ecole des Sciences Criminelles.

Rzepka, L., Ottmann, J., Freiling, F., Baier, H., 2024. Causal inconsistencies are normal in windows memory dumps (too). Digital Threats: Res. Pract. 5, 1–20.

Sandvik, J.P., Årnes, A., 2018. The reliability of clocks as digital evidence under low voltage conditions. Digit. Invest. 24, S10–S17.

Scanlon, M., Breitinger, F., Hargreaves, C., Hilgert, J.N., Sheppard, J., 2023. Chatgpt for digital forensic investigation: the good, the bad, and the unknown. Forensic Sci. Int.: Digit. Invest. 46, 301609.

Schneider, J., Eichhorn, M., Dreier, L.M., Hargreaves, C., 2024. Applying digital stratigraphy to the problem of recycled storage media. Forensic Sci. Int.: Digit. Invest. 49, 301761.

Strom, B.E., Applebaum, A., Miller, D.P., Nickels, K.C., Pennington, A.G., Thomas, C.B., 2018. Mitre att&ck: design and philosophy. In: Technical Report. The MITRE Corporation.

Studiawan, H., Sohel, F., 2021. Anomaly detection in a forensic timeline with deep autoencoders. J. Inf. Secur. Appl. 63.

Sunde, N., 2022a. Constructing Digital Evidence: A Study on How Cognitive and Human Factors Affect Digital Evidence. Ph.D. thesis. Universitetet i Oslo, Institutt for kriminologi og rettssosiologi.

Sunde, N., 2022b. Strategies for safeguarding examiner objectivity and evidence reliability during digital forensic investigations. Forensic Sci. Int.: Digit. Invest. 40, 301317.

Turner, P., 2024. The remapped/reallocated sector conundrum. presentation at dfrws us 2024. URL. https://www.dfrws.org/wp-content/uploads/2023/07/turner-remappedsectors.pdf.

UCO, 2024. Uco project issue tracker. https://github.com/ucoProject/UCO/issues.

van Beek, H., 2018. A forensic visual aid: traces versus knowledge. Sci. Justice 58, 425–432.
Vanini, C., Hargreaves, C.J., van Beek, H., Breitinger, F., 2024. Was the clock correct? exploring timestamp interpretation through time anchors for digital forensic event reconstruction. Forensic Sci. Int.: Digit. Invest. 49, 301759.
Vömel, S., Stüttgen, J., 2013. An evaluation platform for forensic memory acquisition software. Digit. Invest. 10, S30–S40.
Webb, H., Fitzroy-Dale, N., Aqeel, S., Piskopani, A.M., Stafford-Fraser, Q., Nikolaou, C., Dowthwaite, L., Mcauley, D., Hargreaves, C., 2024. Responsible ai in policing. In: Proceedings of the Second International Symposium on Trustworthy Autonomous Systems, pp. 1–5.
Wickramasekara, A., Breitinger, F., Scanlon, M., 2024. Exploring the Potential of Large Language Models for Improving Digital Forensic Investigation Efficiency, 19366 arXiv preprint arXiv:2402.
Wu, T., Breitinger, F., O'Shaughnessy, S., 2020. Digital forensic tools: recent advances and enhancing the status quo. Forensic Sci. Int.: Digit. Invest. 34, 300999.