Forensic Science International: Digital Investigation 37 (2021) 301174

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

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

Reliability validation for file system interpretation

Rune Nordvik [a, b, *], Radina Stoykova [a, c], Katrin Franke [a], Stefan Axelsson [a, d], Fergus Toolan [b]

[a] Norwegian University of Science and Technology, Norway
[b] Norwegian Police University College, Norway
[c] University of Groningen, the Netherlands
[d] Stockholm University, Sweden

ARTICLE INFO

Article history:
Received 26 January 2021
Received in revised form
29 April 2021
Accepted 3 May 2021
Available online 25 May 2021

Keywords:
Digital Forensics
Validation
Reliability
Reproducibility
File systems
Reverse engineering
Black-box testing

ABSTRACT

This paper examines current best practices for Digital Forensic (DF) tool and method validation in the context of file system interpretation for digital evidence. In order to meet the legal and scientific requirements in criminal procedures file system (FS) reverse engineering (RE) is a necessity. Currently, there is no standard procedure for reliability testing of FS RE. Ideal validation requirements exist, but they are on high-level and practical implementation is missing. In this paper we propose a formal reliability validation procedure for file system reverse engineering, documenting the forensic process, including the tools used, ensuring reliability and reproducibility of the method and the results. The procedure is based on legal and scientific criteria and tested against file system reverse engineering methods. It is applicable to all types of reverse engineering methods in digital forensics.

1. Introduction

File systems are one of the richest sources of user activity information, potentially keeping track of every file created, modified, copied or deleted during the entire life span of the device. The majority of digital evidence is found within the file system (Bhat et al., 2020). Access and correct interpretation of the data structures are of core importance for the investigation of criminal cases that involve digital data. The European Network of Forensic Science Institutes (*ENFSI*) emphasizes that analysis of the file system is fundamental for "determining the construction of all files contained within it, and the investigation of provenance of any identified data" (European Network of Forensic Science Institutes (ENFSI), 2015). Similar to all file systems is that they organise files within directories. All file systems use some data structure to point to the location of files on media and store data in file allocation units called clusters or blocks (Kent et al., 2006).

In criminal cases, the investigators, prosecution, defence lawyers, and the court depend on accurate results from tools to interpret FS structures. However, validation and verification of the accuracy of file system interpretation by tools and examiners is still underdeveloped. The reason for this is the proliferation of proprietary or open-source file systems in different devices, each with unique features and data structures, which are described, updated, and improved for practical use and not for digital forensics purposes. Moreover, closed-source commercial digital forensics tools seldom disseminate how they interpret the structures to produce their results, or how the algorithms or methods were implemented (Bhat et al., 2020).

Digital forensic practitioners and academics expressed concerns about the lack of scientific validation in digital forensics (Casey, 2019) (Hughes and Karabiyik, 2020) (Jones and Vidalis, 2019), while the reproducibility crisis in the field was recognized by standardization and governmental bodies worldwide (US President's Council of Advisors on Science and Technology, 2016) (Council of the European Union, 2016). Several legal scholars called for a digital forensic expert accreditation (Henseler and van Loenhout, 2018) (Kwakman et al., 2011) and discussed the absence of clear legal rules for an evidence reliability assessment and the danger of professional bias (Sunde and Dror, 2019) to the

* Corresponding author. Norwegian University of Science and Technology, Norway.
E-mail address: rune.nordvik@phs.no (R. Nordvik).

https://doi.org/10.1016/j.fsidi.2021.301174

disadvantage of all parties in the criminal trial (Risinger, 2018). The rapid advancements in digital forensics render a lot of existing validation schema outdated (Kloosterman et al., 2015), alter reproducibility studies (Horsman, 2019b) (Tully et al., 2020), disturb accuracy testing in digital forensics (Hughes and Karabiyik, 2020), and in the subsequent court evaluation (Saks and Koehler, 2005).

Highlights of the study:

- Dual-tool verification is a procedure largely employed by law enforcement to prove reliability of DF-tool results. However, the procedure is not reliable given the fact that: *(i)* libraries and functionalities are reused in different tools; and *(ii)* the procedure relies on the erroneous assumption that different programmers do not make the same errors which is disproved in a study of N-version programming (Knight and Leveson, 1986).
- Current peer-reviewed, published articles on FS interpretation are often used by law enforcement as guidelines. However, such articles are documenting mainly results and are sparse with information on the accuracy of the scientific method used when performing FS RE, the quality and quantity of testing data sets, and tools employed to reach those results.
- Peer-review cannot alone provide quality assurance for digital forensics in a law enforcement context. Instead, formal validation procedures, which are agreed and enforceable, are required.
- Tools are seldom error-free and the same holds true for validation procedures. What however is essential to be changed in practice, is that the scientific approach must be documented in terms of technology, methodology, and application level. This ensures that *(i)* reliability and accuracy can be tested by others with the same or different techniques; *(ii)* and others can build on the knowledge and reuse it.

Therefore, each law enforcement entity must perform tool verification or rely on reverse engineering to test the FS interpretation accuracy. Previous published test reports by other law enforcement entities may also be used if applicable to describe the FS interpretation accuracy for the same tool version and the same FS version, but only if the test includes the minimum documentation for reliability validation as proposed in this paper. Validation and verification are defined below, and are based on ISO/IEC 27041 (ISO/IEC, 2015).

- Validation is confirmation that the user requirements for a specific purpose (intended use) have been fulfilled, and the validation is carried out on processes (e.g., demonstrating the suitability of a selected algorithm for a specific process).
- Verification is confirmation that a product conforms to specified requirements, and these requirements are related to the product specifications (e.g., showing that the algorithm is implemented correctly).

We adopt these definitions and propose a validation methodology which can demonstrate "the accuracy and reliability of a process" (Hughes and Karabiyik, 2020) for a specific purpose (here FS RE). In addition, we identify customer requirements for law enforcement needs. Tools are evaluated only as far as they are part of, or the complete forensic method.

As examined by Horsman (2019a) there are three types of validation - following previous case work precedents, following existing published works, validation via testing. However, only validation via testing is meeting the requirements for scientific rigor (Gross and Mnookin, 2003).

This paper is part of a cross-disciplinary action research[1] that combines digital forensics and legal considerations in order to identify a new framework for digital forensics reliability testing. Here we are aiming to understand issues of scientific validation of file system interpretation by addressing the following questions:

- What are the reliability validation requirements for FS interpretation in digital forensics?
- Does existing literature for FS RE comply with those requirements?

This paper is organized into seven sections. *Section 1* has introduced the validation challenges of tools and methods, and then defined the difference between validation and verification. In order to identify scientific criteria for validation testing for FS RE methods, first in *Section 2* we present a background on the importance of reverse engineering as a law enforcement evidence examination and validation technique; and in *Section 3* the existing standards regarding process and tool validation are examined. In *Section 4* we delineate concrete requirements for each criterion in order to construct a formal validation model. In *Section 5* we used the validation model to evaluate several file system reverse engineering methods and to establish if the methodology is sufficiently documented, and we summarise to which degree they meet our proposed validation requirements. These results are further discussed in *Section 6*. *Section 7* summarises the paper.

2. Reverse engineering of file systems

Reverse engineering has recently been defined as a "method for finding out how something works, how it is assembled or what the functionality is" (Årnes, 2018, p. 267). When it comes to reverse engineering of file systems it is necessary to understand the inner workings of the file system in order to port it to another operating system platform (NTFS was ported to Linux, and is based on reverse engineering (Carrier, 2005, p. 274)). However, this kind of reverse engineering is not for law enforcement purposes. Law enforcement need to understand which actions trigger a change in a structure; from where it can be triggered; when the change is performed; how the structure is updated; if other events can give identical changes. This knowledge can be used to create additional alternate hypotheses and null hypotheses for testing. It is also important for Law Enforcement Agencies (LEAs) to understand the meaning of fields in a structure, if the fields are mandatory for a functional file system, etc. This is because anti-forensics (AF) techniques can hide within file systems, and the research of Bhat et al. (2020) show that many current DF tools fail on detecting AF. Scrutinising structures on a low level in a hex viewer will show these file system AF attempts, but it requires the investigator to understand the structures.

Developers who develop a file system driver based on file system reverse engineering also need to know which actions trigger a change in a file system structure, but they may be able to design a fully functional, alternative file system driver which updates for instance the access timestamp differently compared to the original file system. From a law enforcement perspective the interpretation of the access timestamp is different if the access timestamp is updated on all access, or if it is only updated the first time a file is accessed after creation, or if it is never updated after creation, or if it is only updated if the volume is less than a particular size. For NTFS on Windows 10 the access timestamp is updated by default if the volume is 128 GiB or less, and not updated if larger than 128 GiB,

[1] Action research is defined as an applied research that focuses on an existing problem, with the goal to improve current practices and desired outcomes. See (Leedy and Ormrod, 2016, ch 10).

but this behaviour may be changed in the Registry or by using the fsutil command (Brink, 2020). Previous versions of Windows update access times differently.

Based on the known structures and their behaviors investigators may, during a digital evidence examination and analysis:

- discover malicious activity, for instance malware hiding in metadata structures.
- understand and interpret the data structure, for example the granularity of timestamps, what triggers update of a timestamp, if there are settings that can impact updating, etc.
- recover user data, for instance identifying signatures in metadata structures that can be used for metadata carving and easy recovery of files.

Reverse engineering can also be used as a validation technique:

- to interpret the FS, by manually assessing the structures and by performing black-box testing. Black-box testing is a technique of testing without having any knowledge of the internal working of the system (Khan and Khan, 2012).
- to verify tool results, using reverse engineering to document structures, and compare with the tool results.
- to overcome tool limitations, since not all tools interpret all relevant metadata structures, or the file system may have been updated with new features after the tool release, or not supported by any tools.
- to cross-examine FS analysis methodology, for instance by experts hired by the defence.

Law Enforcement user requirements may differ based on the intended use of the reversing process, including any reverse engineering tools or methods used as part of the process. For each intended use there should be confirmation that assures the reversing process complies with the user requirements (ISO/IEC, 2015). The user requirements may differ between jurisdictions, however, there are a few similar requirements. The most important LEA user requirements are that the results are accurate and reliable, and that risks of errors and bias are mitigated (Sunde and Dror, 2019). In addition the reverse engineering process should be documented to allow it to be peer-reviewed or challenged.

3. Validation literature review

Currently, there are no internationally agreed standards for DF tool and method validation and court requirements for reliability of digital evidence may vary strongly among jurisdictions. Several authors expressed concerns about the lack of sufficiently large, centralized validation efforts and the lack of reproducibility studies in digital forensics (Hughes and Karabiyik, 2020) (Page et al., 2019) (US President's Council of Advisors on Science and Technology, 2016). For instance, Hughes and Karabiyik (2020) focus on the issue of tool validation. Horsman (2019b) noted several problems, including a lack of reference data sets, a lack of sufficiently large, centralized validation efforts, an inability to exhaust all imaginable testing scenarios, a lack of reproducibility studies, the inability to distinguish tool errors from user errors, rapid changes in the digital ecosystem, and the increasing expectations of examiners. Page et al. (2019) wrote: “In the context of achieving and maintaining quality standards, DF suffers from a number of governance issues. The discipline has consistently debated the omission of regulatory standards and entry requirements to be able to practice within this area.”

Therefore, we further examine guidelines by standardization bodies and academia in order to define ideal validation criteria.

3.1. Guidelines

3.1.1. NIST

The US National Institute of Standards and Technology (NIST) pioneered the development of Computer Forensics Tool Testing (CFTT) (National Institute of Standards and Technology, last accessed 22.11.2020) which consists of a specific methodology to demonstrate the reliability of forensic results, to identify potential errors, and at the same time to support admissibility of evidence. CFTT can test groups of related issues together as case studies, which are focused on the forensic function, rather than a specific type of tool. However, for now only limited digital forensic tool functionalities are tested. CFTT is a complex and time-consuming procedure which is unable to keep pace with updates, versioning and changes in forensic tools or underlying software. To speed up the process NIST also created a procedure for federated testing, where the test data is provided for each laboratory to test it separately and share the results with NIST. This creates a knowledge database about common issues with tools in different versions, the data from file systems they cannot represent or they miss. However, NIST has brief descriptions of common file systems with references to the FS vendor documentation (Kent et al., 2006). This suggests, that NIST trusts DF tool vendors to have correctly implemented the file systems they claim to support, and they also describe that the technical information provided by the tool developers are not necessarily tested (National Institute of Standards and Technology, 2020a). The CFTT tool catalogue provides information on supported FS, which is described as technical parameters reported by developers (National Institute of Standards and Technology, 2020a). It also defines some specifications for data recovery and file carving tools.

NIST states the need for every laboratory to test their own tools and to report the results back to the standardization body. NIST does not provide guidance on which validation method can be used in practice or what are the requirements for reliability testing of different tool functionalities. Currently, there is no NIST specification for validation of file system's interpretation. Moreover, NIST performed tests for most common mobile acquisition tools routinely used by non-skilled investigators to acquire data from FS (Department of Homeland Security (DHS), 2019, Table 3b p. 12–14). The results show that the big forensic suites, such as XRY, UFED Cellebrite, etc, are interpret “not as expected” data from the most common instant messaging apps, social media services, and file systems (National Institute of Standards and Technology, last accessed 22.01.2021). Not as expected means that the mobile forensic application failed to return expected test results. This study shows that testing of tools can not rely on the tool developer, therefore, verification and validation should be performed by an independent body at European or international level. Such an independent validation is recommended by both legal-forensics scholars (Edmond, 2012) and researchers in black-box testing (Wilsdon and Slay, 2006) (Flandrin et al., 2014). Consequently, LEAs must also validate their DF tools and methodology for the specific purpose of criminal investigation. In addition, tool results from commercial tools must always be verified by using a known base truth image.

3.1.2. ENFSI

The European Network of Forensic Science Institutes (ENFSI) aims at harmonization of digital forensics standards in Europe and have published a best practice manual for the forensic examination of digital technology (European Network of Forensic Science Institutes (ENFSI), 2015). ENFSI pointed as a major problem to the lack of "any [internationally] recognized quality standards" for digital forensic processes and systems, resulting from the lack of transparency (Council of the European Union, 2011).

When it comes to lab equipment and digital forensic tools, processes need to be validated that they are meeting the requirements for the intended use by the LEAs, since software and equipment have a limited lifespan. Data sets used to test software and hardware should be meaningful, appropriate and proportionate to the test requirements. It is also an ENFSI requirement that authority should be sought in advance before publishing to avoid disclosure of intellectual property (European Network of Forensic Science Institutes (ENFSI), 2015). Further, they describe that the peer-review should be performed by other competent persons, who have not participated in the analysis. The ENFSI manual recommends proficiency testing of the lab staff. When it comes to reverse engineering of third-party software, the ENFSI guidelines advise the lab to consult their legal department to assess if it is allowed, or ask for permission from the supplier. The manual requires validation and verification of forensic methods and tools according to predefined requirements for file systems analysis (European Network of Forensic Science Institutes (ENFSI), 2015, p. 60). Those requirements demand that RE techniques to analyze a known input with the result output should always be possible (European Network of Forensic Science Institutes (ENFSI), 2015).

3.1.3. National standardisation efforts

In US and China accreditation of digital forensics laboratories and validation specifications are developed from national regulators (Guo and Hou, 2018). In the Netherlands Register for Court Experts (NRGD) certifies court experts in digital forensic according to six fields of expertise, which are computer, software, database, multimedia, device, and network forensics (Netherlands Register Grechtelijk Deskundigen, 2016, §5). Those developments are symptomatic for proliferation of general validation requirements in the absence of specific testing environments and scenarios. Moreover, only in UK the forensic regulator requires obligatory accreditation of digital forensic labs (Forensic Science Regulator, 2020), while in most countries' standards are not legally enforced. The tendency to regulate at national level might result in a proliferation of standards and different quality levels for digital forensics. Consequently, the lack of standard validation procedures during all stages of the digital forensic process, and moreover their implementation to document the scientific approach with its objective measurements as well as the assumptions and interpretations made, impose future legislative and scientific challenge.

3.1.4. Standardisation enforcement

The UK forensic science regulator is the first independent body with power to enforce obligatory implementation of ISO/IEC 17025 (ISO/IEC, 2017) for lab accreditation and network forensics accreditation (The United Kingdom Forensic Science Regulator, 2020). The Regulator also issued the Good Practice Guide Forensic Readiness, which requires mandatory digital forensics capabilities for governmental organizations. However, several studies in the past years expressed concerns that ISO/IEC 17025 is the incorrect vehicle for regulation of standards in DF (Page et al., 2019) (Jones and Vidalis, 2019) (Sommer, 2018). The identified reasons for this are the absence of clear technical requirements within digital forensics service providers and their reluctance to disclose how they fulfill customer requirements (Marshall and Paige, 2018). Validation is reduced as the methods are embedded into use and practitioners fail to review, verify and/or justify changes to the methods, such as software updates (Tully et al., 2020). Most techniques have not satisfied the criteria of known error rates and there is lack of resources and data sets for testing (Jones and Vidalis, 2019). Moreover, the ISO standard does not state which methods in practice can meet those requirements or what is considered sufficient testing. Despite those drawbacks, the standard is currently the only attempt for international harmonization of quality control measures in digital forensics. Importantly, ISO has a requirement to document validation testing (ISO/IEC, 2017, 5.4).

3.2. Current validation procedures

In the past, testing was mainly focused on verification of tools (Lyle, 2010) and black-box testing (Horsman, 2018; European Network of Forensic Science Institutes (ENFSI), 2015), while methodology and application level validation was largely missing. Only recently, validation research emphasized the need for more robust validation methods. Hereafter, we discuss the manifold limitations of DF tools, and also of some RE testing methods such as dual-tool verification and black-box testing in general, in order to justify the need of a formal reliability validation procedure. Friheim's survey (Friheim, 2016) showed that 73% of DF practitioners had found errors in forensic tools at one stage or another, but only 63% verified results in their last case given time and resource limitations. This verification was either performed by using an alternative tool or a manual method, however Friheim (2016) assumes the use of alternative tools. Carrier argued (Carrier, 2002), that most digital forensic file system analysis tools show the files and directories that were recently deleted and, sometimes, can recover them. These tasks were not part of the original file system specification and there is no standard method of performing them. The paper further argues that developers must release their source code if it is used to generate evidence. If a developer is unwilling to do so, then it should be known ahead of time so that it can be a factor when purchasing an analysis tool.

3.3. Digital forensics tool testing limitations

The list below summarizes limitations with accuracy and testing of both open- and closed-source DF tools identified in the literature. Further explained are the DF tool's limitations and the need of a more robust methodology for file system interpretation.

- Underlying FS features remain unsupported by DF tools for significant periods of time (Horsman, 2019b). Even if a DF tool claims to support a particular FS, some features in the FS may be unsupported.
- Validation results become rapidly obsolete due to versioning/updates in all software (Horsman, 2018) (Doyle, 2019)
- Limited versions/functionalities of the same tool are tested, which therefore cover limited scenarios (Horsman, 2019b)
- Testing is time and resource consuming Horsman (2019b). This is discussed by Friheim (2016) stating that "using a larger, commercial tool [(X-Ways, EnCase, FTK, etc.)] for verification can be both too costly and time consuming, whereas the smaller tools available can require too much interaction". Even where code can be accessed for analysis, it is likely that a practitioner would have neither the time or resources to effectively scrutinise its structure for error validation (Horsman, 2018).
- Reuse of libraries/functions and common errors by programmers make dual-tool verification unreliable. This is demonstrated by Friheim (2016) where he emphasizes the

importance of not using shared libraries for dual tool verification.
- Not all DF tools report errors or inaccuracies when parsing file systems (Nordvik et al., 2019). For instance, the investigators should get a tool warning if a file system volume is not parsed/interpreted and it should be logged by the tool.
- Most DF tools can not interpret correctly the file systems when anti-forensic attacks are present (Bhat et al., 2020, Table3)

Given the lack of proof and verification that a tool treats all input data in the same manner, does not omit any data, processes everything according to the forensic objectives, and does not serve personal or corporate interests (Stoykova and Franke, 2020), the evaluation of the results is based on trust. The tool producers are either unable or unwilling to provide information about how they capture customer requirements, let alone disclose what those requirements are (Marshall and Paige, 2018) (Tully et al., 2020). Since results of these tools are used by law enforcement, investigators and courts have to trust that the digital forensic software/hardware was created accurately (Carrier, 2002) (Marsico, 2004) (Patel and Ciardhuaéin, 2000).

3.4. Limitations of dual-tool verification

Dual-tool verification means that a black-box study is performed by using two or more digital forensic tools and then the results from each tool are compared to establish the accuracy. These tools can be both hardware or software based, however we are focusing on software. Friheim (2016) explains that dual-tool verification can only be used if it is "performed by comparing the outcome of different tools which use different libraries, engines and methods for interpreting the same sets of data" (Friheim, 2016). For example, Sleuthkit and Autopsy are both built upon the same set of tools. However, even if the tools do not share code or functionalities, in case they give different results – it will be hard to understand which result is false. Potentially, both tools might be inaccurate if "shared flawed code libraries have been used" (Horsman, 2019b).

3.4.1. Programming errors

Dual tool verification is equal to the idea of N-version programming (Avizienis, 1985), where $N = 2$. In N-version tools more than one copy of a tool/function is deciding what action to perform based on voting. However, using the copies of the same tool do not handle design faults (Avizienis, 1985). To avoid such design faults, Avizienis (1985) suggests using different implementations from the same specifications (similar to N-version programming), and uses different similar results to make a decision of action.

However some implementations based on different designs, may also produce similar errors, for instance based on not checking input values (interpretation faults) (Avizienis and Kelly, 1984). The reasons for these similar errors may be:

- The detail level of the specifications
- Communication between teams in N-version programming
- Using the same compiler
- Using the same programming manual

When digital forensic investigators use one similar software to verify the result of their first software, this is similar to 2T/1H/2 dS, where 2T = two times and 1H = one hardware, and 2 dS = two diverse software. However, in this case we will have no majority voting, since there are only two tools. Increasing this to 3-version or 5-version allows voting, but we will have examples where the majority is wrong. However, even knowing which of the tools are wrong require that we know that the input is the same, and that the investigator already knows the expected output. Avizienis and Kelly (1984) suggest using acceptance tests for each tools, but they also observed errors that were not detected by the acceptance tests.

Another resource issue is that digital forensic investigators need to have multiple digital forensic suites to verify the interpretation results of the file system under investigation.

Knight and Leveson (1986) have performed experiments where 27 different versions (N = 27) independently developed based on the same specification and by different programmers, both novel and experienced, and graduated from different universities, do have correlated errors. They conclude that N-version programming often are based on the assumption that errors are independent, and this assumption is proved incorrect. Therefore, systems that use N-version programming which base their reliability analysis on the assumption on independent errors may not be as reliable as previously believed.

3.4.2. Too much trust in dual-tool verification

Independent errors are also assumed in dual-tool verification. Different tools which implement the same or a similar feature are assumed to not present the same errors when they are developed independently and tested on the same input data set. We argue that dual-tool verification is similar to N-version programming, especially when focusing on similar tool features, and that correlated errors may happen. Therefore, dual-tool verification should not be used as a technique to measure the reliability of the results of another tool.

3.4.3. The popularity of a tool is no measure for quality assurance

Marshall and Paige (2018) describe that even if a large number of practitioners have been using a particular digital forensic tool, this does not mean it is validated (meet user requirements) or verified (are built correctly, and produce the correct results). The techniques or methods could be questionable and secret. Carrier describes that tool acceptance is not equal to procedure acceptance, and he argues that open source tools meet the standard requirements better than closed-source tools (Carrier, 2002). Considering the limitations of dual-tool verification, properly designed "black box" studies of the success or failure of practitioners under different test conditions can yield useable data bearing on the reliability of expert results, which can fill the gap between no formal reliability data and the much more difficult task of generating DNA-like statistical systems (Risinger, 2018). However, black-box studies are inefficient for algorithm and implementation testing (Khan and Khan, 2012). Algorithm and implementation testing require more white-box testing where the source code is scrutinised (Khan and Khan, 2012).

3.5. Dataset availability

Quality, quantity, and availability of data sets for testing in digital forensics ensures that the results are generalisable, reliable, and reproducible. Currently, NIST is the only standardisation body which introduced a large-scale black-box study (National Institute of Standards and Technology, 2020b) where participants are provided with synthetic digital evidence to answer questions that might arise in a criminal investigation with the aim "to document and consolidate information supporting the methods used in forensic analysis and identify knowledge gaps.".

Garfinkel's dataset is one of the largest, and most commonly used for forensic testing (Garfinkel et al., 2009). No similar European initiative exists and given the EU data protection regulation even existing data sets must undergo heavy anonymisation and data minimisation scrutiny which might defeat the purpose.

Moreover, peer-reviewed digital forensic articles are used in digital investigations and are established as a reliability criterion. Validation limitations are related to file system being tested only on a small or non-representative data set. A study by Grajeda et al. (2017) concluded that only 3.8% of the digital forensic researchers actually released their data sets, while the majority of available data sets were synthetic and only around 1/3 originated from real-world data sets.[2] The authors created a database repository with links to data sets available for digital forensic testing, however, some of the links do not correspond to an actual or accessible data set. The study concluded that many researchers prefer not to share their data sets given the lack of platform for publishing, high-volume testing data, but more importantly expressed were data protection and intellectual property concerns.

4. Method

This paper is focused on the FS analysis phase of digital forensics; therefore, questions of acquisition or encryption are not addressed here. It is assumed that the FS data is a result of lawful acquisition in accordance with forensically sound procedure. The Council of EU interpreted that sound digital forensics procedures must reflect the "state of art of science and technology" (Council of the European Union, 2011). It is considered that process or method are forensically sound if they adhere to established digital forensics principles, standards, and processes (Årnes, 2018, p. 13). Many file systems are proprietary, patented or a combination of open and closed source. Reverse engineering is necessary because the vendor documentation is either missing, treated as a trade secret, or is not useful for law enforcement needs. Therefore, such methods require formal reliability validation.

We have justified the need of reverse engineering for validation of file system interpretation, and we have examined the state of the art regarding validation criteria in academic literature and relevant guidelines from standardisation bodies. Since the criteria is on high-level, we further propose for each criterion specific validation requirements.

4.1. Assumptions

Further, assumption is made that reverse engineering methods are a necessity for interpreting closed source FS correctly, and many law enforcement agencies and DF tool vendors rely on the results of reverse engineering.

4.2. Daubert

Our validation model is primarily based on and elaborates further on the Daubert criteria (United States Supreme Court, 1993–1999). The Daubert standard for forensic science was formulated by the US Supreme Court. It became internationally influential for developing better reliability standards in forensics (Jasanoff, 2005). Daubert requires the forensic theory or technique to be: *(1)* tested, *(2)* peer-reviewed, *(3)* generally accepted in the scientific community, *(4)* account for error rates, *(5)* within the examiner's expertise. We identify limitations for implementation of this criteria in digital forensics related to the lack of procedures and lack of standards to produce the information needed for Daubert evaluation (Stoykova, 2021). The identified limitations are as follows:

Testing

- unclear amount, quality, quantity, or type of data needed for validation (Hughes and Karabiyik, 2020)
- as already examined, dual-tool verification may fail due to reuse of libraries/functionalities (Jones and Vidalis, 2019) (Marshall and Paige, 2018)
- test scenarios do not cover all functionalities of tools (Horsman, 2019b)
- lack of time and resources results in lack of quality standards (Horsman, 2018) (Horsman, 2019a)

Peer-review

- what type of peer-review is acceptable (Tully et al., 2020)
- no common understanding of what makes the reviewer an expert (Marsico, 2004)
- in DF the rate of change is faster than the time required for peer-review (Horsman, 2018)
- reference to a peer-reviewed method in the DF report is insufficient, because it does not explain how the method was applied in the particular case (Carrier, 2002).

General acceptance

- most unclear, even irrelevant requirement (Carrier, 2002). This requirement poses challenges because a general acceptance of a closed-source tool having a specific feature is something else than acceptance of the actual procedures or algorithms implemented to realise this feature.
- proliferation of methods and practices in many areas of the digital forensics, where none are considered a standard (Marsico, 2004) (Sremack, 2007) (Arshad et al., 2018)
- no existing program can demonstrate the foundational validity of digital forensic tools, nor provide an examiner or a laboratory the resources needed to perform a comprehensive validation study of a particular implementation of a tool or method (Hughes and Karabiyik, 2020).

Error rates

- lack of methodology to evaluate error rates in DF (Marsico, 2004)
- lack of reporting of errors/bias (Jones and Vidalis, 2019)
- inability to distinguish tool errors from user errors (Hughes and Karabiyik, 2020) (Horsman, 2019b)

Expert skills

- requirements for DF specialists vary among jurisdictions (Henseler and van Loenhout, 2018) (Kwakman et al., 2011)
- competence and impartiality is often assumed by the court (Gross and Mnookin, 2003) (Edmond, 2016)

The process of technology, methodology, and application level validation in our model aims to overcome those limitations.

[2] Real-world data sets refers to data generated by a user(s) during usual activity and interaction with the system. Synthetic data set is a data set generated under controlled testing conditions for instance computer-generated, simulation or experiment (Grajeda et al., 2017).

4.3. Framework for reliable experimental design - FRED

We base the method level of our approach on FRED (Horsman, 2018). FRED includes four stages: planning, test of environment, implementation, and evaluation. In the planning stage outlined is the importance of a hypothesis formulation and preparedness for "unforeseen issues, events and newly acquired knowledge from preliminary findings". The Test setup documentation requires control parameters and objective measurements. Importantly, during the implementation stage the author discusses the need for suitable testing data sets, but also documenting the examiner interaction with the data set and tool as a set of actions or inputs. In the final evaluation stage, the examiner must be able to trace back the steps taken, repeat or modify some examination steps by simultaneously ensuring reproducibility. The aim of FRED is to ensure the results of a digital forensic investigation are reliable and derived from sound research. However, a limitation of the model is that it is focusing on verifying results, not on a validation for a specific use. No consideration is given to how a DF examiner can trace back the identified steps of the experiment in a tamper-proof way. To enable such evaluation, practitioners must document and implement validation measurements on the design stage of the experiment.

4.4. Proposed validation model

The proposed validation model describes the information that needs to be documented in order to assess the validity of reverse engineering of file systems for investigation purposes.

A recent study developed the theoretical background and a reliability-challenges taxonomy that explains in detail and supports the selected structure as a generalisable framework for documenting any digital forensic process for any type of validation (Stoykova and Franke, 2021). The reliability validation framework is not guaranteeing that the quality standards are fulfilled, but maps minimum documentation to enable reliability validation testing to make the forensic process accountable and testable. Here we summarize the components of this general framework and adapt the model specifically for FS RE. It is applicable to all types of reverse engineering methods in digital forensics.

The core validation criteria for file system reverse engineering is the method and tool used, the testing setup, and the examiner work. A validation process and minimum documentation for each of the criteria is defined on technology, methodology and application level. The advantage of the model is that it overcomes limitations of current technology level testing (e.g. dual-tool verification, black-box testing), by developing a validation framework and criteria for the methodology and application level validation. This allows more robust and complex methods such as RE to be validated in the daily work.

This structure has the advantage to specify requirements for a validation and to concertize them for a practical application and documentation in contrast to the Daubert and FRED high-level criteria. FRED is focused only on test setup validation, while here other factors and levels are considered. The proposed model also avoids some of the limitations of the Daubert criteria. Identified tool and method testing limitations are addressed by proposing a formal validation model including specifications of the testing data set and the examiner work. Peer-review drawbacks are addressed by verifying if peer-reviewed articles contain the information needed for peer-review validation. By selecting only one type of methodology, file system reverse engineering, we aim to evaluate if the established practice can meet forensic validation requirements. This validation framework considers that on a technology, methodology and application level different errors and expert skills needs to be reported. The framework can inform examiners of the minimum documentation required for validation and be used as a template for their day-to-day work. Although the test here is on file systems reverse engineering the model can be reused and extended for other methods and tools as well.

4.4.1. Technology level

On technology level validation documentation must provide a proof that a tool is treating all the input data in the same way, does not omit any data and processes everything according to the forensic objectives (Stoykova and Franke, 2020). In this validation framework, tool is understood as the specific functionality of the automated setup which is employed in the methodology. This may include open, semi-open, or closed source forensic tools such as commercial software, in-house tools and scripts. The tool documentation must include its version, configuration, relevant algorithms and implementation. In commercial or other closed source tools the algorithm and implementation are fixed. In bigger tools a specification of the concrete function used is important. Tools must be able also to report errors in output. Reference to previous validation and verification testing and stating known errors reports e.g. underlying system/software interpretation limitations; bugs in the version and tool's ability to report errors in output is required. The technology level validation answers the *what* questions in the 5WH schema (Ieong, 2006).

4.4.2. Methodology level

Validation of the method is an "assessment of whether a standardized sequence of steps, often employing digital forensic tools, leads to a reliable result" (Hughes and Karabiyik, 2020). Documentation on previous work can serve as a guidance for validation but must not be considered correct. As a preliminary work it may include a reference to peer-reviewed reverse engineering or file system interpretation papers, established practice or examples from previous work. Such a reference can point to the limitations of a previous work and the changes and novelties introduced with the new method. It might include an information about patents, drivers, previous FS RE, or other available file system information used in the method. Methodology level documentation must include a description of the file system, the test setup and the test data set. Minimum documentation of the files system under investigation is related to FS name and version, OS name and version as well as drivers version and implementation. Every new testing data set requires new test setup description on methodology level. FRED (Horsman, 2018) can be used as a guideline to describe an experiment or test setup. The description of the testing data set must be specified as a synthetic or real world data set with its quality and quantity measurements. In this context, quality is how suitable the data sets are for the experiments performed during reverse engineering. For instance if the file system only includes the root directory and only allocated files, then it can not really be used for experimenting on how deletion of a file will affect the metadata structures identified. If the creator has manually inserted file content in unallocated space, but the file system metadata structures are missing, then this is not applicable for understanding the file system. The quantity means how large the sample size is based on the use cases it is meant for, for instance the number of allocated files and directories, the number of unallocated files and directories, the number of different types of files; regular files, links, sockets, etc. In order for the experiment to be reproducible, the testing data set(s) must be available, either as a synthetic or real-world data. Although synthetic data is not as representative as real-world data, it has the advantage that it will include the base truth if the creation of the data set is thoroughly documented. A real-world data set is not necessarily representative

by the fact that it represents the usage of a particular user, and it creates its own issues with how to manage personal data. Knowing the base truth enables computing error rates. It is important to state precision and recall results on a methodology level in order to describe reliability of the results, since the results from a reverse engineering of file systems may be used by DF tools that automate the file system parsing, and may be relied on in the court proceedings. The assumption that the examiner inevitably makes is that the file system will act equal based on the same input no matter if it is synthetic or real-world data. The methodology level answers *how* questions from the 5WH.

4.4.3. Application level

In a day-to-day work it is the responsibility of the examiner to ensure that on application level the method and tool worked correctly and as intended in the specific case (Hughes and Karabiyik, 2020). The examiner skills to perform FS RE can be justified with reference to certification or other proof of domain and topic specific knowledge, competence or experience. Given the fact that digital forensic methods have an empirical nature, on application level the tools and methods must be selected to fit best to the case-specific forensic task. This includes documentation of subjective measurements such as the scope of the analysis, hypothesis, assumptions, and expert knowledge used in the case. This can be executed according to guidelines, templates, or standard operation procedures (SOPs). The examiner interaction with the automated setup must be transparent as well. This includes parameterizations of the tool by selecting features or setting control parameters. Documentation on application level must include the justification reasons for selecting a particular method, algorithm or features. The examiner further must describe confidence level in probabilistic reasoning and perform strict separation of facts and inferences from facts (Gross and Mnookin, 2003) (Casey, 2018). The application level validation answers *why* and *who* questions.

5. Results

In this section we examine established guidelines and papers on file system reverse engineering methodology. Even though some of the papers are not peer-reviewed, they are considered as guidelines. This decision is motivated by the expertise of the examiner, the multiple references to the paper, the limited literature available on RE validation, and the limitations of peer−review processes examined in Section 4.2. We selected these papers based on their relevance to reverse engineering of file systems for law enforcement purposes, and we restricted it to file systems used on personal computers. First of all they should cover a closed-source file system where the main metadata structures are unknown, or only partly known. Further, they should describe a relevant interpretation of these metadata structures for law enforcement purposes. We could have included NTFS, but chose not to do this because the reverse engineering was performed to create an alternative driver for Linux by the Linux-NTFS project (2005). Carrier (2005) based his work on this previous reverse engineering, and discussed how the discovered structures could be interpreted for law enforcement purposes. However, his work does not describe reverse engineering.

We are basing our assessment of reverse engineering papers on our proposed validation model in Table 1, which is developed based on the proposed validation model in Section 4.4, and not all of these criteria were known by the authors of the examined papers. We are not assessing the quality or identifying reliability of results presented in the papers, we only assess if the information found enables validation of the processes used for FS RE. The overall results are summarized in Fig. 1, Fig. 2, and Fig. 3.

5.1. Reverse engineering of ExFAT

Hamm (2009) describes low level structures of the ExFAT file system. He is not claiming that this is reverse engineering.

We consider this testing as one of the early attempts on reverse engineering for digital forensic purposes. It is also considered as an independent result, since it is not performed by law enforcement, but by an employee of Paradigm Solutions.

5.1.1. Technological level

Hamm (2009) has used hex editors to view structures which are not described. He does not specify which tools were used for reverse engineering. The algorithm used for testing is not described. However, we assume that he did not build a file system interpretation tool, and therefore the ability to report errors in output are not applicable in this case. Previous errors, validation or verification reports were not described. He does not meet any of the criteria listed in the technology level in Fig. 1.

5.1.2. Methodology level

Hamm (2009) does not state the setup of his test experiments. He describes using patents, his own observations, and extensive testing. Further, he describes the file system. However, how this testing was performed is not documented. He does not specify any data sets, except for showing a few hex dumps. Since there is no description of data sets, there is also no information about base truth. Limitations of the work are not mentioned.

5.1.3. Application level

From an application perspective, selection of parameters are not listed. The uncertainty of results or confidence level are not described. Hamm (2009) does not describe his knowledge, skills or competences (accreditation, certification of specific knowledge). Further, no description of an investigation scope, hypotheses testing, assumptions were found. He does not try to separate what is facts or inferences. He does not state guidelines or standard operating procedures (SOP) used. The results are presented as offset tables describing important structures, and their interpretation. He did not develop a tool and therefore no source code is mentioned.

5.1.4. Validation summary

We are missing too many of the validation criteria necessary to validate the work of Hamm (2009).

5.2. Reverse engineering of APFS

Hansen and Toolan (2017) reverse engineered the APFS file system for investigation purposes. At the time of reversing there was no detailed information available about the low level structures, and the digital forensic tools available did not support this FS.

5.2.1. Technological level

It is unclear, if Hansen and Toolan (2017) used any tools to test or perform experiments. They documented only several screenshots from hex editors.

5.2.2. Methodology level

However, the authors (Hansen and Toolan, 2017) have described which version of the APFS file system they interpret. They claim that they used reverse engineering and intensive testing, but do not reference to any peer-reviewed method or established practice. How reverse engineering or testing was performed is not documented. The only data sets available are found in the hex dumps used as examples in the publication. There are no description of the

Table 1
Validation model.

Minimum Documentation		
Technology level	Methodology level	Application level
Tool type, name, version Tool features Functions description Algorithms and implementation Prior validation/verification results Known errors reports Tool's ability to report errors in output	Reference to peer reviewed method (limitations); established practice; previous work Patent/FS information used File system description Experiment/Test setup Test data set description Quality and quantity of testing data set Synthetic or real world data Base truth is described	Description of analysis scope Guidelines/SOPs reference Hypothesis/Assumptions/Alternatives/Limitations stated Interaction with tool (Parameterization – feature selection) Justification of method, algorithms and features selection Confidence levels Examiner (Competence/Experience/Topic-specific knowledge) Assessment of tool results Precision and recall Separation of facts and inferences Description of results Source code available

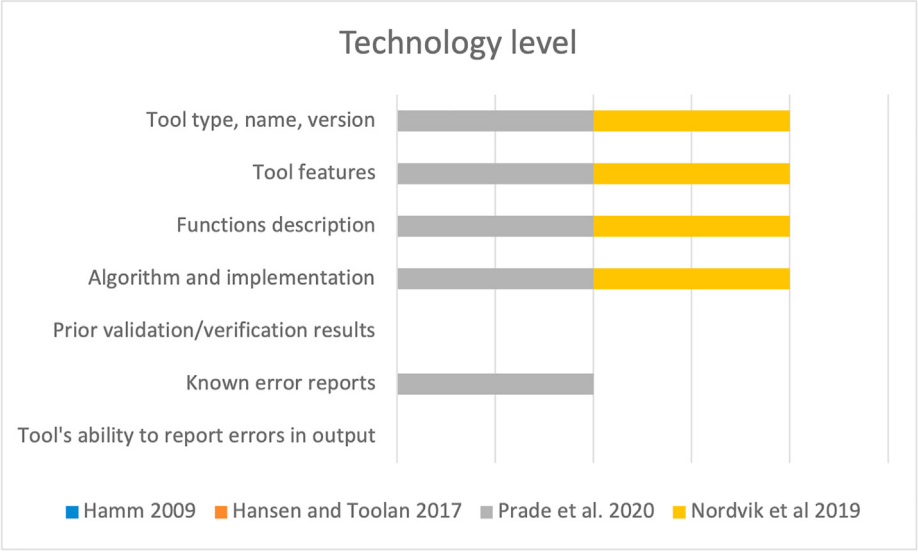

Fig. 1. Result of four guidelines/papers related to Reverse engineering of file systems. Identifying Technology level requirements identified within these papers.

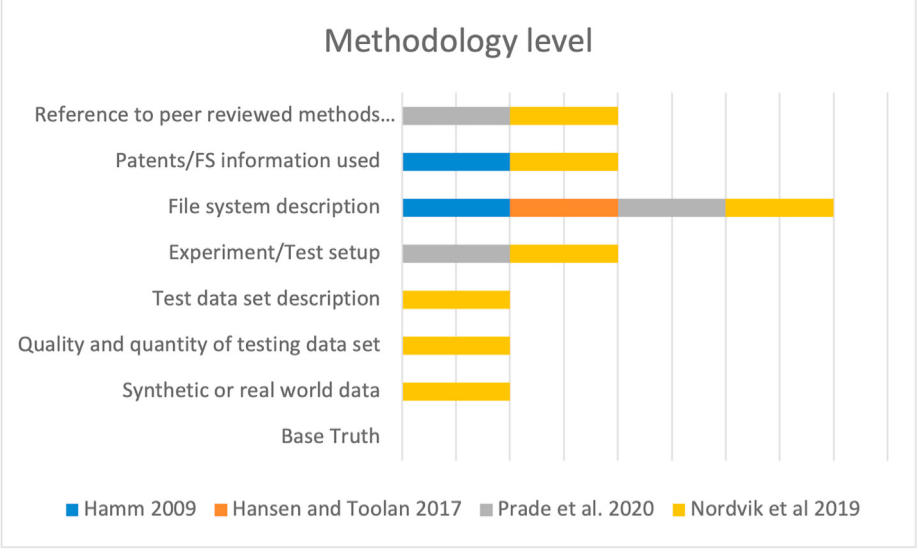

Fig. 2. Result of four guidelines/papers related to Reverse engineering of file systems. Identifying Methodology level requirements identified within these papers.

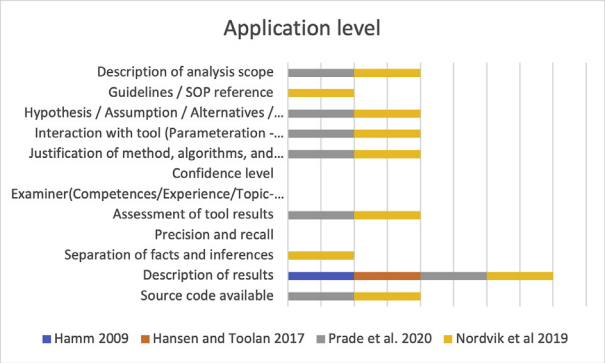

Fig. 3. Result of four guidelines/papers related to Reverse engineering of file systems. Identifying Application level requirements identified within these papers.

data sets used. There is no description of the quality or sample size of the data. It is not described if the data is from a real-world or a synthetic data set.

5.2.3. Application level

The only indications we find about analysis scope is that the analysis is not complete, but sufficient to parse most important structures. From an application perspective nothing is described about feature selection. Hansen and Toolan (2017) do not describe information about the knowledge, skills or competences of the researchers e.g. accreditation, certification of specific knowledge. Hypotheses testing is not described. We can not address the confidence level of the results, or separate facts from inferences.

5.2.4. Validation summary

We are missing too many validation criteria to validate the work of Hansen and Toolan (2017).

5.3. Reverse engineering of ReFS - I

Nordvik et al. (2019) were the first to perform peer-reviewed reverse engineering of the Resilient File System (ReFS). They focused on identifying metadata structures, and to explain their meaning in a digital forensic context. They also introduced new methods for metadata carving and for file recovery. Their contribution enables investigators to verify the results of other DF tools that implement support for ReFS.

5.3.1. Technology level

Nordvik et al. (2019) list the use of tool type, name, versions, for instance using EnCase v.8, which where used as an initial start of the research when focusing on ReFS v1.2. The authors compared the results from EnCase with the interpretation of the metadata found on the storage.

They describe that EnCase has support for ReFS v1.2, which is the tool feature utilised. They also describe details about other tools they used during testing.

The algorithms or implementation of EnCase is not published, but they describe that EnCase should have the feature of parsing ReFS.

For automation of testing purposes, Nordvik et al. (2019) developed a prototype tool. The prototype was open source and made publicly available in order to make it comply with the Daubert criteria, and to allow further development for supporting future ReFS versions. The prototype tool is documented, since the source code is included. Here tool type, name, version, features, algorithm, implementation are included. There are no existing tool error reports (it is a prototype for parsing the file system), and the tool itself does not support reporting errors. The tool only supports ReFS v1.2, and was used for automation of testing.

5.3.2. Methodology level

The authors (Nordvik et al., 2019) describe the file system based on information available from Microsoft.

Previous literature from other FS reverse engineering attempts were included, since knowledge of previous file system may contain relevant and similar artefacts.

They describe how experiments were performed, for instance how the testing of hypotheses were performed. The description could have been more detailed, describing each experiment.

The data sets used were published, allowing researchers to perform the same experiments on the same data, enabling others to verify results.

They do not describe the exact base truth of the data set, and how the data set was created. However, since the data sets are made available, we can see that they used synthetic data sets.

5.3.3. Application level

Nordvik et al. (2019) describe what kind of analysis they perform, but also what is out of scope. The FRED framework was used as a guideline, at least for creating experiments in order to verify tool results. Limitations are described, for instance that the real name of the structures are unknown because debugging symbols did not include the names. They also describe that not all possible use cases are included, and that the result is only valid for the file system versions described.

Interaction with tools (features) are described where applicable, for instance that the *format* command did not have another option than formatting using 64 KiB clusters for ReFS v1.2.

The authors competence, experience or knowledge are not described, and they do not describe their confidence level.

The prototype tool was used to automate testing of identified structures, and results were assessed.

The prototype is not tested for accuracy or reliability, and there are no precision and recall computed.

It seems like most of the results are described based on inferences based on experiments and hypotheses testing. However, they also show that part of the description of the volume boot structure is taken from published material from Microsoft.

5.3.4. Validation summary

This is the paper that meets most of the requirements in the proposed model. It does not describe known error reports, because they did not exist at the time. The prototype developed does not produce errors in output. They did not include information about the base truth and the quantity/quality of the data set. The authors do not describe the researchers competence, knowledge or experience.

However, since the data sets are available it should be possible to find this information, and since the name of all authors are published, their competence, experience can be gathered using open source intelligence or by contacting the authors.

They did not perform any computation of error rates. Further, the confidence level is not described. However, the prototype tool can not be validated before error rates are computed.

5.4. Reverse engineering of ReFS - II

Prade et al. (2020) describe many of the ReFS structures on a high level, and they developed a Sleuthkit module for parsing the ReFS file system. They also developed a tool that is able to carve for more deleted files than their Sleuthkit module can find.

5.4.1. Technology level

The authors (Prade et al., 2020) describe features and some of the functions of the developed tool. The algorithm and implementation are described on a high abstraction level. There are no previous validation/verification of similar tools, and there are no error reports. However, they do describe limitations of their tools, which may work as an error report. They do not describe how their tool handles errors.

5.4.2. Methodology level

Prade et al. (2020) refer to previous research in the domain, but they do not describe how they found all the structures described, and all the experiments performed to support their findings. They do not state if they used patents or other sources to find this detailed information. The experiments for the reverse engineering are not described, however, the experiments to test their tools are documented. They report their data sets, but the data sets do not seem to be available. They describe and justify the use of file system actions of what they assume is realistic, meaning the data sets are synthetic.

5.4.3. Application level

The authors (Prade et al., 2020) do not describe hypotheses or alternatives, but they do describe limitations. They state that they used the fls command of sleuthkit with the option to list only allocated files, which is interaction with the tool. They justify the use of their tools, and the source code is available on GitLab. They do not describe their competences. They assess their tool results, but not to a degree that precision and recall can be computed. When they interpret the ReFS structures it is difficult to separate facts or inferences. They do not specify a confidence level of their tool results.

5.4.4. Validation summary

We have a lot of information that can be used to validate their tools. However, we can not really validate the description of the ReFS metadata structures.

6. Discussion

Even if a paper addresses all criteria in the proposed validation model, this does not mean that the paper is validated, it means only that it can be validated if it fits the customer requirements. However, if a paper is missing important criteria, then the paper can not be validated. In these cases LEAs need to try to find answers from the authors or other sources, or they have to repeat the research on their own data sets and test the reliability of the results.

Each of the criteria need to be assessed. Since validation is all about meeting the customers requirements, the same method can be valid for a security company performing incident response, but not for LEAs when acquiring all relevant data for investigation purposes. It is out of scope for this paper to describe the needs of each stakeholder, therefore, we focus on the law enforcement needs when investigating artifacts found in a file system. Identified requirements for law enforcement purposes are:

- accurate and reliable results based on scientific principles. Investigators can no longer assume that the tools are correctly specified, implemented, and tested (Marshall and Paige, 2018).
- a description of how the results were found, since the investigator should not just trust the tool.
- a description of weaknesses and strengths of proposed methods, for instance by publishing error rates (Garfinkel, 2010).
- efficient validation techniques are needed to decrease the large backlog of cases (Scanlon, 2016).

Law enforcement have to answer the 5WH (who, what, when, where, why and how) questions (Shinder, 2002). From the papers assessed, we can conclude that the main focus has been to disseminate the results of their reverse engineering, and not to describe all information necessary to validate the method. Therefore, we will discuss our proposed validation model below.

6.1. Tools or method

6.1.1. Technology perspective

Including information of all tools used when performing reverse engineering is important in case that other researchers want to repeat the experiments in order to see if they get the same results. If the tool is built in-house and not made available, it will not be possible to repeat the experiments. It is important to include the version of the tools used, which make it possible to repeat the experiment using the same tool versions.

Commercial or open source is a discussion which is important when assessing the validity of results. Intellectual property rights may negatively impact the possibility to assess the reliability of the tool results, without using other methods. If the source is available, it does not mean the results are reliable, but it means an assessment of the source code can be performed. Even if the file system is open-source and information about it is available, LEA still need to learn and find documentation on how to interpret it. Correct interpretation of tool results could take much time of testing and documenting which far exceeds the limit of the peer-review paper publication.

The authors assessed (Hamm, 2009; Hansen and Toolan, 2017; Nordvik et al., 2019; Prade et al., 2020) publish information on how only partial structures can be interpreted and rely that others will build upon it. Usually what is published are detected tool inaccuracies or misinterpretation. Horsman (2019b) points out that due to insufficient testing “tools for the parsing of a file system may focus on displaying file and folder content to a user and inaccurately interpret metadata”.

Digital forensic tool vendors do not sufficiently document tool information for validation. There is lack of proof and verification that a tool is treating all the input data in the same way, does not omit any data and processes everything according to the forensic objectives, and does not serve personal or corporate interests.

In some cases the patents or other documentation may be available, and this was the case in the documentation of the ExFAT file system (Hamm, 2009). However, investigators can not just trust the documentation as facts.

6.1.2. Methodology perspective

In order to repeat an experiment all the information about the test setup needs to be described. It is all the tools and hardware used in the experiment that is important to document, this also includes versions of the operating system, installed service packs, drivers, libraries, etc. The standard operating procedures or guidelines followed should be documented. Also how the tools were selected, for instance based on previous error reports. The use of tools, methods should be based on previous validation or verification testing. However, just because a tool or method is validated for a particular purpose, it may not be validated for the intended use case.

6.1.3. Application perspective

If tools or a method are used, it may require different parameters to work as intended for a specific purpose. The algorithm used should also be specified, because different algorithms may give different results. Any errors found during the experiment should be documented. There is also a need to document why a specific

algorithm was used, and also why specific features were selected. Typically, each result is either true positive, false positive, true negative or false negative. These results need to be known, or at least a conditional set of the results should be known. It is important to measure precision (how accurate are the artifacts found) and recall (how many relevant artifacts are included in the results).

Normally, the degree of precision impacts the degree of recall. High recall often means a lot of false positives, and therefore a low precision. High precision often means a lot of false negatives, ergo a low recall.

When results from tools are not verified we can experience errors. This can be exemplified in relation to hash functions.

Hash functions are traditionally considered a strong, reliable authentication and integrity preservation method, which is often accepted as scientifically valid without testing. A cryptographic hash function is a non-reversible mathematical function which takes any amount of data as an input and returns a fixed size string as an output (Synopsys Editorial Team, 2015), which is often referred to as digital fingerprint. The acquisition tool is making a bit by bit copy of the device on the new media in raw.dd format or compound (E01) with a hash digest MD5, SHA1, SHA2, SHA3 etc. Some errors in the hash functions can have impact on the reliability of the evidence. Firstly, MD5 and SHA1 functions have been found to be vulnerable to certain attacks but they are still in use. Furthermore, in all new versions of hash functions the authors are listing the errors and bugs they have fixed. For example, a recent report (Qt, 2020) stated that "in Qt versions before 5.9, when asked to generate a SHA3 hash sum, QCryptographicHash actually calculated Keccak." Ergo, all hashes calculated with v.5.9 were inaccurate due to a programming error. This also means all tools that used QT and the QCryptographicHash SHA3 function had this error.

6.2. Training data sets

6.2.1. Technology perspective

Different hardware technologies may impact the data set before acquisition. For instance SSD disks may implement a garbage collection and remove data from the file system before or during acquisition, even when a write blocker is used (Gubin, 2018). Therefore, it is important to describe the device/storage medium. If one reused existing data sets, the original description of the data sets should be referenced.

It is difficult to repeat the experiment and get the same results if the data sets are missing. Real-world data set can be important for generalisation, including the selected sample size (quantity). The more real the data is, the harder it is to know the complete base truth. In order to measure precision and recall we need to know the base truth of the data set used. We advise to use synthetic data sets for file system reverse engineering, because the content of files is not important in order to experiment with the metadata structures. However, the size of a file may impact how metadata is structured.

6.2.2. Methodology perspective

The experiment needs to be clearly defined in order to allow other researchers to repeat the study or the same researcher to reproduce it, and obtain the same results.

6.2.3. Application perspective

When performing reverse engineering of file systems it is important to describe the scope of the reverse engineering (analysis scope). Current file systems have many structures, and not all of them are relevant for investigation purposes. The aim of the reverse engineering may not be a complete reverse engineering of the file system, but may only include interpretation of metadata describing files. It is necessary to know the scope, in order to understand if the results can be included in a validated process. For instance, if the focus of the reverse engineering is on detecting malware hiding in Alternate Data Streams in allocated MFT records. Then the use case is not applicable if malware is hiding in another type of NTFS attribute. Depending on the analysis scope, and the utilized methodology, the feature selection may impact the reliability of results. This is specially important if the researcher is using a machine learning algorithm in order to assess the file system structures (Nguyen et al., 2010). It also depends on the accuracy of known structures from previous FS reverse engineering.

6.3. Examiner

6.3.1. Technology perspective

The researchers need to have competence to perform reverse engineering of a file system. Formal education, certification, and specific domain knowledge will describe if the researcher is qualified to perform the FS reverse engineering. However, currently there are no agreed ideal competence requirements.

6.3.2. Methodology perspective

Just because a paper is referring to a peer-reviewed method, it does not guarantee that scientific methodology was used or all the results are reliable. It is just as important to address how the paper/method was reviewed, and what competence the reviewers have in the domain. Some journals have page restrictions, which may force the authors to prioritise what content to include. This may negatively impact the possibility to assess the reliability of the results presented in the paper. Not all journals inform who reviewed the paper, which may avert the assessment of the reviewer's competence. The researcher needs to follow scientific methodology such as defining hypotheses, and testing null hypotheses. All assumptions should be described, including obvious ones. The use of technical terminology should be explained or referenced to help the reader understand the publication.

6.3.3. Application perspective

Even if the tool validation and method peer-review were done correctly, this cannot guarantee that the examiner will apply them correctly or the results will be reliable in the new case. Modifications of the method and parameterization of the tool must be documented. In addition, the method, feature and algorithm selection must be justified according to the concrete forensic task and data set structure and characteristics. The researcher should describe uncertainties in results, and how confident they are about the reliability of the results. If possible, facts should be separated from interpretation and inference.

7. Conclusion

- What are the reliability validation requirements for FS interpretation in digital forensics?

We have proposed a novel validation model that considers the technological, methodology, and application perspectives. These requirements are shown in Table 1 and justified in Section 4.4.

- Does existing literature for FS RE comply with those requirements?

We assessed papers of file system reverse engineering for digital forensic purposes. The results show that researchers' main focus is on disseminating results. The selected papers and current practices do not describe important validation criteria, which we have

discussed and justified as documentation minimum requirement. More recent papers from 2019 to 2020 included more validation criteria than the older literature, which is a positive development in the domain. However, Prade et al. (2020) documentation was mostly related to the tools they developed, not their reverse engineering of file system metadata structures. Common challenges identified are quality and quantity of data set for testing and computing error rates.

There is the need of meeting legal and scientific reliability standard in digital forensic examination for court proceedings. This will greatly depend on a systematic approach for validating the interpretation of a file system. Information about the requirements in the proposed validation model can be gathered from authors, practitioners or other sources, and even by performing reverse engineering. In these cases our validation model can be used as a template. Such a template encompasses only the minimum documentation requirements, and more elaborated versions can be developed in further research. Future work can be dedicated to establishing an independent EU body to specialise in tools and methods validation, and to promote the use of purpose-validated DF tools and methods among law enforcement agencies.

The proposed validation model is also applicable when interpreting applications that utilise their own metadata structures for storage purposes. For instance, most current applications (Apps) on a smart phone use SQLite databases, which need to be interpreted, not only from the perspective of the SQLite structures, but also when interpreting the undocumented meaning of the tables created by the App developer; column names, type of content, and how these are updated based on user actions within the App. The only change needed in the validation model is changing the criteria for the file system (FS information used, File system description) with the criteria for the database or database system. We can conclude that for any undocumented, or partly documented, metadata structures this model is applicable.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The research leading to these results has received funding from the Research Council of Norway programme IKTPLUSS, under the R&D project “Ars Forensica - Computational Forensics for Large-scale Fraud Detection, Crime Investigation & Prevention”, grant agreement 248094/O70.

References

Årnes, A. (Ed.), 2018. Digital Forensics: an Academic Introduction. John Wiley & Sons Inc, Hoboken, NJ.
Arshad, H., Jantan, A.B., Abiodun, O.I., Apr, 2018. Digital forensics: review of issues in scientific validation of digital evidence. Journal of Information Processing Systems 14 (2), 346–376.
Avizienis, Kelly, 1984. Fault tolerance by design diversity: concepts and experiments. Computer 17 (8), 67–80.
Avizienis, A., 1985. The n-version approach to fault-tolerant software. IEEE Transactions on Software Engineering SE- 11 (12), 1491–1501.
Bhat, W.A., AlZahrani, A., Wani, M.A., 2020. Can computer forensic tools be trusted in digital investigations? Science and Justice 61 (2), 198–203.
Brink, S., 2020. How to Enable or Disable NTFS Last Access Time Stamp Updates in Windows 10. https://www.tenforums.com/tutorials/139015-enable-disable-ntfs-last-access-time-stamp-updates-windows-10-a.html.
Carrier, B., 2002. Open Source Digital Forensics Tools: the Legal Argument.
Carrier, B., 2005. File System Forensic Analysis. Addison-Wesley Professional.
Casey, E., Mar. 2018. Clearly conveying digital forensic results. Digit. Invest. 24, 1–3.
Casey, E., Nov. 2019. The chequered past and risky future of digital forensics. Aust. J. Forensic Sci. 51 (6), 649–664. https://doi.org/10.1080/00450618.2018.1554090 publisher: Taylor & Francis _eprint:
Council of the European Union, 2011. Conclusions on the Vision for European Forensic Science 2020 Including the Creation of a European Forensic Science Area and the Development of Forensic Science Infrastructure in Europe. https://www.europeansources.info/record/council-conclusions-on-the-vision-for-european-forensicscience-2020-including-the-creation-of-a-european-forensicscience-area-and-the-development-of-forensic-scienceinfrastructure-in-europe/.
Council of the European Union, 2016. 6078/16 Draft Council Conclusions on the Way Forward in View of the Creation of an European Forensic Science Area. https://data.consilium.europa.eu/doc/document/ST-6078-2016-INIT/en/pdf.
Department of Homeland Security DHS Science and Technology Directorate S&T by the Office of Law Enforcement Standards of the National Institute of Standards and Technology, 2019. Test Results for Mobile Device Acquisition Tool. UFED InField Kiosk v7.5.0.875. https://www.dhs.gov/sites/default/files/publications/testresultsnistmobiledeviceacquisitiontool-ufedinfieldkiosk_v7.5.0.875.pdf.
Doyle, S., 2019. Quality Management in Forensic Science. Elsevier : Academic Press, is an imprint of Elsevier, London, United Kingdom ; San Diego, CA, United States, oCLC, on1022794523.
Edmond, G., Jan. 2012. Is reliability sufficient? The law commission and expert evidence in international and interdisciplinary perspective (Part 1). Int. J. Evid. Proof 16, 30–65.
Edmond, G., Jan. 2016. Legal versus non-legal approaches to forensic science evidence. Int. J. Evid. Proof 20 (1), 3–28 (publisher: SAGE Publications Ltd).
European Network of Forensic Science Institutes ENFSI, 2015. Best Practice Manual for Forensic Examination of Digital Technology. https://enfsi.eu/wp-content/uploads/2016/09/1._forensic_examination_of_digital_technology_0.pdf.
Flandrin, F., Buchanan, W., Macfarlane, R., Ramsay, B., Smales, A., 2014. Evaluating Digital Forensic Tools (DFTs). /paper/Evaluating-Digital-Forensic-Tools-(DFTs).-Flandrin-Buchanan/4feaa848491f9bec6fb275612f9dcf8d627f2a37.
Friheim, I., Aug. 2016. Practical Use of Dual Tool Verification in Computer Forensics. Ph.D. thesis, University College Dublin. https://www.researchgate.net/publication/324784023_Practical_use_of_dual_tool_verification_in_computer_forensics.
Garfinkel, S.L., 2010. Digital forensics research: the next 10 years. Digit. Invest. 7, S64–S73 (the Proceedings of the Tenth Annual DFRWS Conference).
Garfinkel, S., Farrell, P., Roussev, V., Dinolt, G., Sep, 2009. Bringing science to digital forensics with standardized forensic corpora. Digit. Invest. 6, S2–S11.
Grajeda, C., Breitinger, F., Baggili, I., Aug. 2017. Availability of datasets for digital forensics – and what is missing. Digit. Invest. 22, S94–S105.
Gross, S., Mnookin, J., Jan. 2003. Expert Information and Expert Evidence: A Preliminary Taxonomy. Articles.
Gubin, A.V., 2018. Write Blockers Are Not Effective with Ssds. https://www.klennet.com/notes/2018-04-16-write-blocking-ssd.aspx.
Guo, H., Hou, J., Jul. 2018. Review of the accreditation of digital forensics in China. Forensic Sciences Research 3 (3), 194–201.
Hamm, J., 2009. Extended Fat File System visited 2020-11-10. https://paradigmsolutions.files.wordpress.com/2009/12/exfat-excerpt-1-4.pdf.
Hansen, K.H., Toolan, F., 2017. Decoding the apfs file system. Digit. Invest. 22, 107–132.
Henseler, H., van Loenhout, S., Mar. 2018. Educating judges, prosecutors and lawyers in the use of digital forensic experts. Digit. Invest. 24, S76–S82.
Horsman, G., Mar. 2018. Framework for Reliable Experimental Design (FRED): a research framework to ensure the dependable interpretation of digital data for digital forensics. Comput. Secur. 73, 294–306.
Horsman, G., Mar. 2019a. Formalising investigative decision making in digital forensics: proposing the digital evidence reporting and decision support (DERDS) framework. Digit. Invest. 28, 146–151.
Horsman, G., Mar. 2019b. Tool testing and reliability issues in the field of digital forensics. Digit. Invest. 28, 163–175.
Hughes, N., Karabiyik, U., 2020. Towards reliable digital forensics investigations through measurement science. WIREs Forensic Science n/a (n/a), e1367, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wfs2.1367.
Ieong, R.S., 2006. Forza – digital forensics investigation framework that incorporate legal issues. Digit. Invest. 3, 29–36 the Proceedings of the 6th Annual Digital Forensic Research Workshop (DFRWS '06).
ISO/IEC, 2015. ISO/IEC 27041:2015 Guidance on Assuring Suitability and Adequacy of Incident Investigative Method. https://www.iso.org/standard/44405.html.
ISO/IEC, 2017. ISO/IEC 17025:2017 General Requirements for the Competence of Testing and Calibration Laboratories. http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/69/66912.html.
Jasanoff, S., 2005. Law's knowledge: science for justice in legal settings. Am. J. Publ. Health 95 (Suppl. 1), S49–S58.
Jones, A., Vidalis, S., 2019. Rethinking digital forensics. Annals of Emerging Technologies in Computing 3, 41–53.
Kent, Karen, Chevalier, Suzanne, Tim Grance, Dang, Hung, 2006. NIST SP 800-86, Guide to Integrating Forensic Techniques into Incident Response. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-86.pdf.
Khan, M.E., Khan, F., 2012. A comparative study of white box, black box and grey box testing techniques. Int. J. Adv. Comput. Sci. Appl. 3 (6).
Kloosterman, A., Mapes, A., Geradts, Z., van Eijk, E., Koper, C., van den Berg, J., Verheij, S., van der Steen, M., van Asten, A., Aug. 2015. The interface between forensic science and technology: how technology could cause a paradigm shift in the role of forensic institutes in the criminal justice system. Phil. Trans. Roy.

Soc. Lond. B Biol. Sci. 370 (1674).
Knight, J.C., Leveson, N.G., 1986. An experimental evaluation of the assumption of independence in multiversion programming. IEEE Transactions on Software Engineering SE- 12 (1), 96–109.
Kwakman, N., Nijboer, J., Keulen, B., Elzinga, H., 2011. Expert registers in criminal cases. In: Governance in Criminal Proceedings. https://www.rug.nl/rechten/congressen/archief/2011/governancemeetslaw/workingpapers/papernijboerkeulen.pdf.
Leedy, P.D., Ormrod, J.E., 2016. Practical Research: Planning and Design, eleventh ed. Edition. Pearson, Boston.
Linux-NTFS project, 2005. Welcome to the linux-ntfs project. https://flatcap.org/linux-ntfs.
Lyle, J., 2010. Verification of Digital Forensic Tools. https://www.nist.gov/system/files/documents/2017/05/08/montana-may-2010.pdf.
Marshall, A.M., Paige, R., Dec. 2018. Requirements in digital forensics method definition: observations from a UK study. Digit. Invest. 27, 23–29.
Marsico, C.V., 2004. CERIAS Tech Report: Computer Evidence V. Daubert: the Coming Conflict. https://www.cerias.purdue.edu/assets/pdf/bibtex_archive/2005-17.pdf.
National Institute of Standards and Technology, 2020. NIST disk imaging. https://www.cftt.nist.gov/disk_imaging.htm.
National Institute of Standards and Technology, 2020a. Computer Forensics Tools & Techniques Catalog. https://toolcatalog.nist.gov/.
National Institute of Standards and Technology, 2020b. NIST to Digital Forensics Experts: Show Us what You Got. First Large-Scale “Black Box” Study Will Test the Accuracy of Computer and Mobile Phone Forensics. https://www.nist.gov/news-events/news/2020/06/nist-digital-forensics-experts-show-us-what-you-got.
National Institute of Standards and Technology, 2021. NIST mobile devices. https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt/cftt-technical/mobile.
Netherlands Register Grechtelijk Deskundigen, 2016. Standards 008.0 Digital Forensics. https://www.nrgd.nl/binaries/Standards%5cDigital%5cForensics_tcm39-82994.pdf.
Nguyen, H.T., Franke, K., Petrovic, S., 2010. Towards a generic feature-selection measure for intrusion detection. In: 2010 20th International Conference on Pattern Recognition, pp. 1529–1532.
Nordvik, R., Georges, H., Toolan, F., Axelsson, S., Sep. 2019. Reverse engineering of ReFS. Digit. Invest. 30, 127–147.
Page, H., Horsman, G., Sarna, A., Foster, J., Jan. 2019. A review of quality procedures in the UK forensic sciences: what can the field of digital forensics learn? Sci. Justice 59 (1), 83–92.
Patel, A., Ciardhuaéin, S., 2000. Impact of forensic computing on telecommunications. IEEE Commun. Mag. 38 (11), 64–67.
Prade, P., GroB, T., Dewald, A., 2020. Forensic analysis of the resilient file system (refs) version 3.4. Forensic Sci. Int.: Digit. Invest. 32, 300915.
Qt, 2020. Qcryptographichash Class. https://doc.qt.io/qt-5/qcryptographichash.html.
Risinger, D., Jun, 2018. The five functions of forensic science and the validation issues they raise: a piece to incite discussion on validation. Seton Hall Law Rev. 48 (3).
Saks, M.J., Koehler, J.J., Aug, 2005. The coming paradigm shift in forensic identification science. Science (New York, N.Y.) 309 (5736), 892–895.
Scanlon, M., Aug, 2016. Battling the digital forensic backlog through data deduplication. In: 2016 Sixth International Conference on Innovative Computing Technology (INTECH), pp. 10–14.
Science Regulator, Forensic, 2020. Codes of Practice and Conduct, Appendix: Digital Forensic Services. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/912389/107_FSR-C-107_Digital_forensics_2.0.pdf.
Shinder, D.L., 2002. Scene of the Cybercrime: Computer Forensics Handbook. Syngress Publishing.
Sommer, P., Jun. 2018. Accrediting digital forensics: what are the choices? Digit. Invest. 25, 116–120.
Sremack, J.C., 2007. The gap between theory and practice in digital forensics. In: 2007 Annual Proceedings in ADFSL Conference on Digital Forensics, Security and Law, pp. 85–94. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.823.8791&rep=rep1&type=pdf.
Stoykova, Radina, 2021. Digital evidence: Unaddressed threats to fairness and the presumption of innocence. Computer Law and Security Review, CLSR-D-20-00299. In press.
Stoykova, Radina, Franke, Katrin, 2021. Reliability validation framework for digital forensics. Journal of Digital Forensics, Security and Law. Manuscript 1737. Submitted for publication.
Stoykova, R., Franke, K., May, 2020. Standard representation for digital forensic processing. In: 2020 13th International Conference on Systematic Approaches to Digital Forensic Engineering (SADFE), pp. 46–56.
Sunde, N., Dror, I.E., Jun. 2019. Cognitive and human factors in digital forensics: problems, challenges, and the way forward. Digit. Invest. 29, 101–108.
Synopsys Editorial Team, 2015. What are cryptographic hash functions? https://www.synopsys.com/blogs/software-security/cryptographic-hash-functions/.
The United Kingdom Forensic Science Regulator, 2020. Method Validation in Digital Forensics, FSR-G-218, Issue 2. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/921392/218_Method_Validation_in_Digital_Forensics_Issue_2_New_Base_Final.pdf.
Tully, G., Cohen, N., Compton, D., Davies, G., Isbell, R., Watson, T., Mar. 2020. Quality standards for digital forensics: learning from experience in England & Wales. Forensic Sci. Int.: Digit. Invest. 32, 200905.
United States Supreme Court, 1993-1999. Daubert V. Merrell Dow Pharmaceuticals, inc., 509 u.S. 579, 1993. The Daubert Criteria Was Further Elaborated in General Electric Co. V. Joiner 522 u.S. 136 (1997), and Kumho Tire Co. V. Carmichaet 526 u.S. 137 (1999). United States Supreme Court.
US President's Council of Advisors on Science and Technology, 2016. Forensic Science in Criminal Courts: Ensuring Scientific Validity of Feature-Comparison Methods. https://obamawhitehouse.archives.gov/blog/2016/09/20/pcast-releases-report-forensic-science-criminal-courts.
Wilsdon, T., Slay, J., 2006. Validation of Forensic Computing Software Utilizing Black Box Testing Techniques. In: 4th Australian Digital Forensics Conference Edith Cowan University, December 4th 2006, Medium: PDF Publisher: Security Research Institute (SRI). Edith Cowan University.