Evaluating Digital Forensic Tools (DFTs)

Flavien Flandrin, Prof William J. Buchanan, Richard Macfarlane,
Bruce Ramsay, Adrian Smales
School of Computing, Edinburgh Napier University, Edinburgh.
Contact email: w.buchanan@napier.ac.uk

1 Introduction

Digital forensics has been developed in a way to other types of forensics. With other forensic sciences, methodologies have often been based on scientific discoveries, and through ad-hoc research [1, 2, 3]. However, due to the rapid growth of digital investigations, scientific processes are now integrated into investigations in order to make digital forensics evidence acceptable in court [4, 3]. Work has also been undertaken to provide ways to help juries understand the value of digital evidence [4]. Robertson defines this the evolution over the next ten years for forensic sciences and illustrates some of the current challenges:

> "New technologies and improved instrumentation will continue to emerge. Forensic science usually has a lag period before these are adopted into the forensic area, and this is to be expected, given the conservative nature of the legal arena. In the past, however, the lag period has been too long. Forensic scientists need to be quicker to recognize the potential applications to forensic problems and they also need to be able to carry out research aimed at helping to interpret what analytical data means." [5]

As digital forensics is still an immature science, legal issues still exist which need to be overcome, such as from Meyers and Rogers [6] who highlighted three main issues faced within computer forensics:

- Admissibility of evidence. In order to ensure that evidence are admissible in court, investigators should follow rigorous procedures. These need to be standardised, or, at minimum, guidelines should be defined. Unfortunately, even closely following recommendations, errors can be made. This is often due to each case being different, and it is not possible to create a manual which can cover all possibilities. For this reason, it is essential that digital investigators develop appropriate skills to get round these problems.
- Standards and certifications. Certifications are a good way to develop investigators' skills. This has is already successfully applied in other computer fields, such as computer security [7]. A small number of certifications – such as the Certified Forensic Analyst [8] certification, which is provided by the Global Information Assurance Certification (GIAV) founded by the SANS Institute – are available. Nonetheless, certifications and standards are not only applicable to investigators.
- Analysis and preservation of evidence.

Beckett [3] stressed that compliance with the ISO 17025:2005 [9] laboratory accreditation is the best way to bring digital forensics the same level as other forensic sciences. The

International Organization for Standardization (ISO) associated with the International Electrotechnical Commission (IEC) created this standard in order to provide laboratories general requirements to carry out, tests, calibrations and sampling. The main requirements are the following:

- Management system
- Document control
- Subcontracting of tests and calibrations
- Purchasing services and supplies
- Service to the customer
- Complaints
- Corrective action
- Preventive action
- Test and calibration methods and method validation
- Assuring the quality of test and calibration results
- Reporting the results

Projects have been carried out by organisations in order to evaluate Digital Forensic Tools. The most well-known project is undertaken by the National Institute of Standards and Technology (NIST) under the Computer Forensics Tool Testing (CFTT) project [10]. Results of these tests are released to the public. The Scientific Working Group on Digital Evidence (SWGDE) and the DoD [11] also assess Digital Forensic Tools, however, results are available only to U.S. law enforcement agencies [12]. It is difficult to understand the reason behind the choice in not release information because computer forensics, as with any other science, is based on information sharing. Even if all results were available, it would not be possible for these organisations to keep up with the fast pace of tool development [13]. In addition, many practitioners rely too much on vendors capability to validate their own tools [3]. Furthermore, Beckett [3] defined that, in order to comply with the standard requirements, laboratories should not rely on testing performed by another organisation.

The risk of failure in Digital Forensic Tools has been proven by different authors. NIST [14] demonstrated that the famous acquisition tool `dd` [15] was not able to retrieve the last sector of the hard drive if it had an odd number of sectors. These results have been confirmed by Kornblum [16]. However, the author explained that the behaviour was not coming from the tool implementation. Instead, he argued that this issue was coming from the `Linux Kernel 2.4`, but not from `Linux Kernel 2.6`, which demonstrates that organisations which validate DFTs can make mistakes. If the results are followed blindly by laboratories, a major issue might arise if errors have been introduced in the testing procedure.

The previous example discussed the issues related to software. However, investigators might have to use other tools which combine hardware and software such as *write blockers*. NIST produced an evaluation methodology for this type of products and evaluated multiple write blockers. Beckett [3] properly explained the risks that a laboratory might encounter if no additional testing is carried out. Each device needs to be tested before it can be used in the field, but a manufacturing fault may be present, or that the device was damaged, for instance, during transport.

Pollitt [17] argued that evaluation methodologies and standards for the digital forensics field were needed. Since then there has been little work done on developing methodology requirements or evaluation methodologies. The creation of certification for digital forensics investigators is a great step forward, but there is a lack of strong evaluation methodologies [6, 13, 18].

2 Existing Methodologies

Most studies on digital forensics focus on the viability of the employed methodologies rather than the tools used to perform the investigation [13]. As a result, little research has been carried out on DFT evaluation and validation, which leaves investigators with few resources to assess their tools.

2.1 Carrier's Model of Abstraction Layers

Carrier [19] was one of the first attempts in defining a evaluation methodology for DFT, and produced a model which focused on the identification and analysis phases of an investigation. It thus excluded the collection and validation parts, but that the abstraction layers are already present in digital investigation. For instance, when data is acquired, it is normally in a raw format, and is extremely complex for a human to handle. Thus, tools have been developed to represent the raw data in an abstracted form. The model (Figure 1) principally targets these tools, and introduces two type of errors:

- *abstraction error* These are generated because of simplifications used to generate the layer of abstraction. ASCII is an example of abstraction layer where nothing is lost, and Carrier designed it as a *lossless layer*. When ASCII characters are stored on a computer system, they are mapped to a numeric value, and no information is lost. Nevertheless, other tools are losing information by moving from one abstraction layer to another. These are called *lossy layer* [19]. An exampleof an abstraction error is an IDSs. When an IDS maps a set of packets to an attack, a margin of error is introduced as the tool cannot know if all the packets are part of the attack. In fact, it is possible to have on error rate for each passage to an abstraction layer to the other. Pan [20] develop a model based on Carrier's one. This model can be used to define the abstraction layers. In addition, the model integrates a solution to produce time flow of the "read" and "write" operations.
- *implementation error*. This is caused by bugs from the design and implementation of the tools' functions [19]. Carrier introduced the concept of a multiple error rate for the same tool.

Figure 1: Carrier's Abstraction Layers [19]

The abstraction model is an interesting concept, however, is fairly complex, where the tester needs to possess extensive digital forensics knowledge of the internals of the tool. This might not be easy as the DFT might be proprietary. Even if it was open source, the tool might be poorly documented. The only option left in such a case is

to read the source code to understand the tool's workings. Such a process is often time consuming and complex, and requires other skills from the tester than purely digital forensics ones. Furthermore, it seems to be a slow and tedious process. For these reasons, this evaluation methodology is not suited to be implemented on a large scale in order to assess a large range of DFTs. Even if it was implemented and used, the complexity of the methodology will be a major issue.

2.2 NIST Standardised Approach of Tool Evaluation

In the Computer Forensics Tool Testing (CFTT) project, NIST developed methodologies to validate a range of forensics tools, initally focusing on data acquisition tools [21, 22] and write blocker [23, 24] (software and hardware based). Figure 2 illustrates the methodology used to assess the tools [10]. When a tool will be tested, the NIST methodology starts by acquiring the tool, with a review of the tool documentation. If this documentation is non-existent, the tool is analysed in order to generate such documentation, and which leads to a list of features along with the requirements for these features, and thus a test strategy.

This methodology is based on rigorous and scientific methods, and the results are reviewed by both of the stakeholders (vendor and testing organization), ensuring a certain level of fairness. However, this is also the major weakness of this methodology, as the time required for the evaluation can be significant. The resources needed to carry out each test does not enable a single organisation to test all tools along with all versions [13]. Thus, by the time the results are publicly available, the version of the tested tool might be deprecated. In addition, the requirements of features might evolve which need to be reflected in the test strategy. Moreover, the time needed to define the requirement of a single function need to be counted in years. NIST has defined standards for string searching tools [25], but since dditional work has been made publicly available. The specifications for digital data acquisition tools are still in a draft version since 2004 [21], and these examples show that this methodology is not viable for law enforcement agencies to rely only on organisations which evaluate DFTs. Some categories of tools commonly used in digital investigation are only not covered, such as *file carving* tools. For these reasons, it is essential for digital investigators to validate DFTs themselves.

2.3 Validation and Verification of Digital Forensics Tools with Reference Sets

Beckett [3] explained that testing may not find all errors of a DFT, due to the fact that a complete evaluation of a product would require extensive resources. The requirements defined by ISO 17025:2005 [9] specifies that validation is a balance between cost, risk and technical possibilities. However, testing should be able to provide information on the reliability of the tool.

Before looking at solutions to validate and verify digital forensic processes, it is essential to define:

- **Validation**. This is the confirmation by examination and the provision of objective evidence that the particular requirements for a specific intended use are fulfilled" [9]
- **Verification**. This is the confirmation of validation with a laboratories tools, techniques and procedures" [3]

The methodology created is represented in Figure 3, and proposes that, in order to validate a DFT, it is essential to know the expected results of the tested *forensic function*

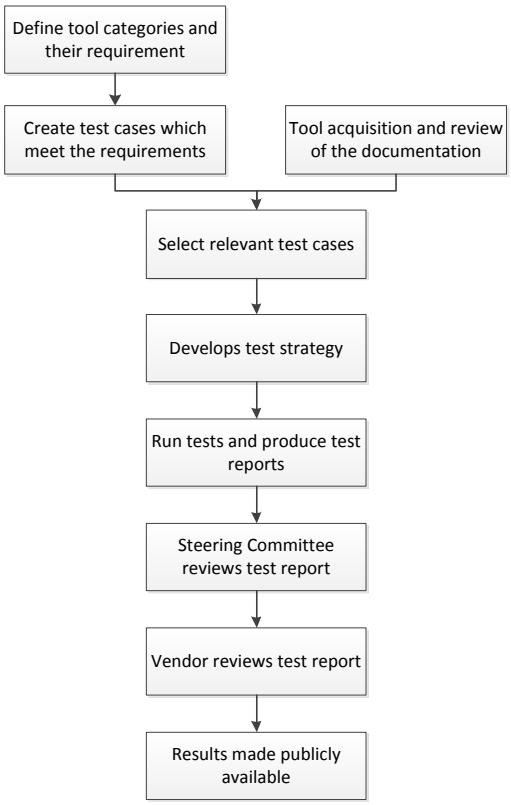

Figure 2: NIST testing methodology [10]

along with its domain. This process is named *function mapping*, and enables them to draw diagrams which represent the different components of the function. This level of granularity permits the extraction of metrics which will be calculated by comparing the expected and obtained results, and allow the comparison of forensics functions which are from the same domain. The expected results are defined by Beckett and Slay as *reference sets*, which enable the evaluation methodology to meet the following requirements:

- **Extensibility:** The reference set is used to model all specifications of a particular function. If new specifications are found, they can be easily added to the reference set.
- **Tool Neutrality:** Any tools which implement forensics functions from the same domain can be evaluated with the same reference set.
- **Tool Version Neutrality:** Same as the previous statement. As long as functions are part of the same domain, the reference set will not be required to be modified.
- **Transparency:** The reference set can be publicly available and anyone can audit it in order to improve its quality

Finally, they argue that if a function fails in a particular scenario, the function should not be invalidated. Instead, the results should be used to correct the tool weakness. Such a methodology can be implemented in the Software Development Life Cycle (Software Development Life Cycle) for vendors. This would help them improving tools' functionality and ensure that there is no loss of quality when these functions are updated. This methodology has been further refined by Guo [26], who mapped the requirements

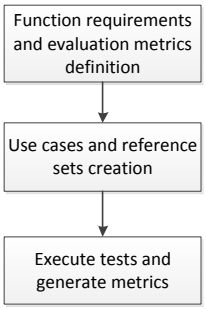

Figure 3: Validation and Verification of DFTs with Reference sets [3]

for *string searching* functions and defined a reference set. They did the same for the *data recovery* function [27, 28], however, they did not perform actual validation and verification on any existing tools.

The presented methodology provides a robust solution to evaluate tools' functions. This approach is the most sensible as a tool might perform well for some tasks and poorly for others. If the tool was rated as a whole, the results would not be useful to anyone. In addition, the definitions of metrics is strongly linked with the tested function. A range of metrics can be defined but they will not apply for each functions. Therefore, it is required to define them for each function. Finally, they propose to use reference sets in order to enable the reproducibility of the evaluation process.

2.4 Black Box Testing Techniques for DFTs

It has been acknowledged that "vendors cannot anticipate all uses in all situations therefore they can only test a finite set of permutations of systems and uses" [29]. It is thus essential to provide the digital forensics community with solutions to cover this lack of testing. For instance, laboratories will have to perform tool testing which cover their own methodologies to ensure that the DFTs are properly evaluated and validated. Wilsdon [13] created a methodology based on *black-box testing* principles, as most Digital Forensic Tools are proprietary. With no access to source code. Without knowledge of the tool implementation, it is not possible to evaluate each function at a low level. The testing regime is composed of six steps as shown in Figure 4. The different steps defined by Wilsdon [13] are:

1. The acquisition of the relevant software need to be documented in order to comply with ISO 17025:2005 requirements. Then, a hash signature of the file is created, so that if someone else wants to validate the results, it is possible to ensure that the same program is tested.

2. Tool's functions need to be identified. This step defines the limits of the test, as only these functions will be tested.

3. In order to validate the tool, test cases need to be generated. These test cases will support *black-box* testing principles. During this step, a *reference set* needs to be acquired or created, such as for the *reference sets* used in [13].

4. Because expected results are known, it is possible to compare them with the results obtained in the previous step. Thus, it is possible to define metrics which will measure the correctness of the tested tool, and the level of expected accuracy can be defined. It is expected that functions will not be perfect, but a given rate of errors might be acceptable, as defined as the *acceptance range*. For instance,

the rate of *false positives* would possibly slow down an investigation, but it would not compromise it. Finally, if a function does not meet the acceptance range, the function and all functions which depend on it should be classified as being incorrect.

5. Tests are performed and results are recorded. Any unexpected behaviour (such as program failure) are also noted, and produced results are compared with the expected results, with resultant metrics. This permits grading of tested functions as "pass" or "fail"

6. Results, reference sets and test cases are made publicly available to the digital forensics community to enable result validation by peers.

This methodology enables anyone to test and validate Digital Forensic Tools, along enable products to be tested in many different environments. As it is not possible for vendors to test all possibilities [13], this methodology provides results closer to reality. Finally, Wilsdon argues that an authoritative body needs to be created to implement the methodology and allocate resources for its application. Furthermore, it is argued that the automatic creation of reference sets is needed due to the level of skill and the amount of time required to produce them manually. The DFT can also be stressed from different angles which is the best solution to detect all weaknesses. Black-box testing has been successful for other type of software [30] and it seems to be a viable solution to test Digital Forensic Tools. Nevertheless, the methodology has never been implemented.

One of the main limitations of this methodology is regarding the "fail" and "pass" results. This classification implies that all tools which "passed" are from the same quality level. Because there is a threshold, it is essential to differentiate a function which barely passed and a function which is nearly perfect. Hence, granularity should be included to take into consideration this aspect.

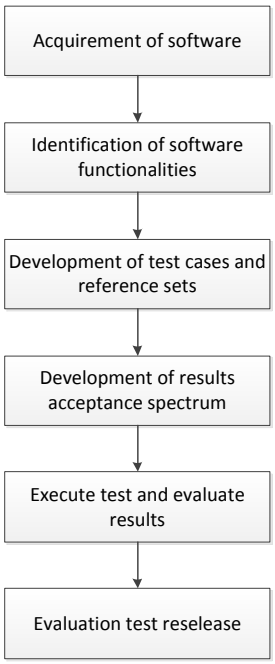

Figure 4: Black-box Testing Methodology for DFTs [13]

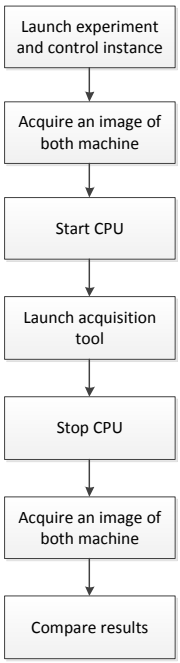

Figure 5: Live DFTs Evaluation with *Pypette* [32]

2.5 Live Forensics Tool Evaluation

Live forensics is often poorly understood by investigators [31], and the evaluation techniques are virtually non-existent. One of the few peer reviewed papers on the subject discusses of the implementation of a framework call *Pypette* [32]. With this live data should be extracted carefully, or the data will be refused as evidence, and they developed a framework which measures the impact of live forensics acquisition tools. In order to do so, a snapshot of the machine-under-test is taken before and after running the acquisition, and the differences analysed for (Figure 5):

- Amount of memory used by the acquisition tool
- Impact on Windows Registry
- Impact on the file system
- Use of Dynamic Link Libraries [32]

The implemented methodology uses virtualisation to effectively control the machine-under-test. In order to take snapshots of the tested machine, the authors created an agent which is present in the system. They also measured the impact of the agent on the system of 0.1%, and it can thus be ignored as a significant overhead. This framework is interesting as it propose a novel way to evaluate and validate live acquisition tools.

2.6 Performance Testing for Digital Forensics Tools

Pan [12] focuses on a method which can be used to evaluate the performance (such as execution speed) of DFTs, and argue that it is not wise to assume that DFT testers will be able to obtain high quality results. This is due to the fact that testers might not be knowledgeable enough or are ill-equipped. They used a partition testing approach as opposed of random testing approach, and define that the complexity of fully tested software with random testing has the complexity of $\mathcal{O}(l^k)$ where k is the number of

parameters where each of them has l values. This can be compared with, partition testing which has a complexity of $\mathcal{O}(l \times \log_2 k)$. For this reason, partition testing is an improved solution as it would take less time to complete.

In order to implement this solution, they chose Orthogonal Array (OA), which is a technique where an array is created covers every solution. Initially different parameters are specified, such as a parameter which is a list of OSs and another is the amount of RAM available. In order to produce fair results, it is required to perform the test with every combination of OS and RAM. This permits to cover all possible cases, however, another problem might appear in the results of the experiments. Pan argued that it was not possible to perform experiments without including errors in it: *outliers*, which is "an observation that lies an abnormal distance from other values in a random sample from a population" [33]. In order to reduce the impact of outliners, a data transformation technique, defined by Taguchi [34], was used with the function $f(x) = -10\log(x^2)$, where x is an observed result is applied to the entire set of results [34]. This dismissed outliers and reduced the impact of suspicious samples. In addition, they created a Theorem 1 to define the maximum number of suspicious samples acceptable.

Theorem 1 *N is the number of rows of the OA. Then, for any integer $N \geq 1$ where $N \times R$ observation withstand the impact of measured and random error, the maximum number of suspicious samples should not exceed*

$$N_c = \min\left\{\lfloor\frac{N-1}{2}\rfloor + \lfloor\frac{N+1}{2}\rfloor \cdot \lfloor\frac{R-1}{2}\rfloor, N \cdot \lfloor\frac{R-1}{2}\rfloor\right\}$$

Finally, Pan defined the following procedure which follows the previous requirements:

- **Step 1.**Execute the OA once.
- **Step 2.**Results are flatten with Taguchi function and suspicious samples are identified. If no suspicious samples are detected then steps until Step 6 are ignored.
- **Step 3.**Compare the number of suspicious samples with the results of Theorem 1
- **Step 4.**R' is the number of observations which has already been made. Perform the tests $R - (R' + 1)$
- **Step 5.**Results are flatten and suspicious samples are identified on the new sets of results.
- **Step 6.**Each row of the OA is tested and results are flatten. Suspicious samples are counted, and if they are higher than the maximum number of results found with Theorem 1, the procedure should be restarted from Step 3.

This procedure supports a high level of automation if the parameters involved do not require human intervention, where testers only need to confirm the presence or absence of suspicious samples which then determine the next step of the procedure. Pan [12] expect to implement an automated solution for this methodology in the future. Nevertheless, the proposed methodology has issues which have been acknowledged by the authors. First, the methodology assumes that a definitive set of parameters can be defined and controllable. However, due to the nature of computing it is possible that software fails in unexpected ways. In addition, it might not be possible to use this methodology for any type of DFT.

3 Digital Forensics Tools and Error Rates

Digital forensics investigators have to follow laws and regulations in the country where they exercise. Nonetheless, a legal decision taken in 1993 by the Supreme Court of the United States had a huge impact on the field [35], where the Dauber [36] trial assessed the admissibility of expert witnesses during scientific testimony of a legal proceeding. Key finding outlines that the conclusions drawn by expert witnesses need to be based on scientific methodologies and defined as *science knowledge* [35]:

1. The theory or technique is testable and have been tested previously.
2. This theory is subjected to peer review and publications. A number of specialised journals and conferences now exists for the digital forensics field, including *International Journal of Digital Investigation.*
3. The potential error rate generated by the technique needs to be known. This statement is around *error rates* is virtually unaddressed at the current time. Only Lyle [35] attempted to identify basic issues of defining *error rates* for DFTs.
4. The technique is generally accepted within the relevant scientific field.

The *error rate* represents the level of uncertainty in the scientific method. For instance, if a sample is analysed in order to find a particular component – called $\mathcal{X}$ – two possibilities exist. The first, $\mathcal{X}$ is present in the sample, and the second, $\mathcal{X}$ is not present in the sample. Therefore, there are four solutions:

1. The sample $\mathcal{X}$ is present in the sample and it has been found
2. The sample $\mathcal{X}$ is present in the sample but it has not been found
3. The sample $\mathcal{X}$ is not present in the sample and it has not been found
4. The sample $\mathcal{X}$ is not present in the sample but it has been found

The first and third solutions are the expected outcome of a test, however, it is possible there is an error. It is these issues which need to be quantified in order to measure the *level of uncertainty* of a method. Two types of errors can occur. The first corresponds to the last solution and it is known as a *false positive*, also called error Type I. The second is illustrated by the second solution, and called a *false negative*, or error Type II [37]. Lyle [35] explained that, at first glance, it is possible to find errors rate for DFTs. The simplest solution is

$$\frac{k}{n} \tag{1}$$

where n is the total number of bits acquired and k the number of bits incorrectly acquired. Nevertheless, the Lyle argued that things are not that simple in the digital forensics world. For him, forensics processes tend to have an error that are *systematic* rather than *statistical* (or statically distributed). The employed algorithm, its implementation and the person who use it, can introduce errors. It is also possible that a tool will only work under certain conditions. For instance, a tool which generates MD5 hash signatures can work properly for text files, but fail with binary files. Moreover, hashing algorithms are complex, and could have a poorly implemented algorithm, with some files with specific content not be process correctly. It might also be difficult to differentiate *false negatives* against standard tool behaviour. Regarding the forensics acquisition tool example, if a disk sector has not been acquired, it can be because of two reasons. The first is that

the tool fail to extract the data which is a *false negative*. However, it is possible that the sector is damaged and cannot be read, and that this is not an error from the tool.

This highlights the risk of generating error rates which cannot be trusted. Lyle [35] observed that the error rate needs to be linked with the condition under which they have been produced. For instance, the version of the targeted OS or the type of hard drive, might have an influence on the results. For these reason, it was concluded that the global error rate for a function or a tool might be meaningless. If error rates are to be generated, all parameters which can have an impact should be identified. For these reasons, it might be required to create multiple error rates for the same digital forensics functions. Nevertheless, it is still possible to satisfy the "spirit of Daubert" [35] by describing the type of failure and triggering conditions of DFTs.

4 Digital Forensics Tools Evaluation Metrics

In order to validate DFTs, it is vital to use a range of metrics for various reasons. First, it will enable the community to compare different tools on an unbiased basis. Secondly, vendors will know which sections of a tool needs to be improved. Metrics need to cover the maximum properties of Digital Forensic Tools in order to provide accurate information on the tool capacity to fulfil certain requirements. This area of DFT testing is not widely discussed as explained in Section 2 and few testing methodologies have been designed, with little research on the definition of metrics to validate Digital Forensic Tools. Ayers [38], though, defined seven metrics to evaluate DFTs which are:

- **Absolute speed** represents the time required by the tool to complete a task.
- **Relative speed** compares the average processing evidence rate against the rate to read data from the original media.
- **Accuracy** is the proportion of correct results.
- **Completeness** represents the proportion of evidence found from the pool of evidence available in the forensics image.
- **Reliability** measures how often the tool is likely to fail during an investigation.
- **Auditability** defines if the results are fully auditable.
- **Repeatability** measures the proportion of tests where the process employed was exactly as specified.

Additional work has been undertaken by Pan and Batten to define metrics who proposed a solution to measure the *accuracy rate* and the *precision rate*. The first represents the amount of evidence correctly extracted from the list of evidence, while the second represent the number of extracted files from the list of files [39]. The authors also propose a methodology to perform performance evaluation of tools [12].

These metrics cover most aspect of DFTs, however, some of them are not detailed enough. For instance, Ayers does not provide a details definition for "correct results", which might be interpreted differently by different people. For some, such as Pan [39], a correct result might be when the retrieved evidence is identical to the original evidence. This can be checked with hashing algorithm such as MD5. Nevertheless, if only one bit is modified during the acquisition, the hash signatures will be different. For others, a correct result is one that can be used in court. For instance, if an illegal image file is extracted from a disk with only one bit which differs from the original evidence, there, the hash signatures are different. It is highly possible that the image

file can still be viewed by users, and the incorrect bit might have little impact on the file content. Furthermore, the extraction process might not retrieve the exact data because of problems which are not caused by the tool. It is possible, also, that the disk is damaged, and the tool would not be able to extract the complete file.

5 Digital Corpora

A digital corpora can be defined as a large structured set of digital data, which can be used to validate techniques which rely on statistics and other mathematical approximation. As these techniques will have been tested against a wide range of publicly known and validated data, the results will be based on scientific grounds [40], and can thus be a core part of testing DFTs. However, digital corpora are often difficult to create, thus some researchers create them from real evidence, but these are often difficult to publicly distribute due to privacy and ethical issues. In addition, it is possible that organisations can set limitations on the sharing of this type of data, such as for formal requests to be made each time digital corpora os shared, even if it does not possesses sensitive information. This administrative burden can slow down experiments and might discourage research.

Without digital corpora, it is not possible for researchers to validate and compare their work. Garfinkel et al. [40] discussed file identification, where each of the cited authors produced metrics on the accuracy of their methodology, and used these metrics to compare their methodologies and draw conclusions. Because none of them used the same data to perform their tests, comparing the results did not make sense as it is not possible to know if the testing data introduce a bias during the evaluation process. Hence, it is not possible for a digital forensic investigator to choose the best methodology to perform this task. Klimt and Yang [41] exposed similar issues regarding their work on automatic email classification.

For these reasons, it is essential to create standard digital corpora. Garfinkel et al. [40] defined four main categories of digital forensics corpora and are defined as the following:

- **Disk Images:** This is the most common type of digital corpora. A complete image of a system's hard drive is use as the source of experimentation
- **Memory Images:** They represent the virtual memory use by computer. As it is been previously demonstrated, they might contains relevant data. Therefore, it is essential to provide such corpora for testing and training.
- **Network Packets:** Represents network traffics which has been recorded.
- **Files:** In most cases, files are the main evidence. Therefore, it is vital to provide files as digital corpora. Files can be used individually or with others in order to produce more complex digital corpora.

Furthermore, Woods et al. [42] specified that realistic digital corpora for education should have realistic wear and depth., where it should look like if users have been using the system for standard activities. In addition, the digital corpora should not only be composed of evidence. The background part – background noise – is essential to add realism to the data. The authors created a set of digital corpora for training purpose only. The digital corpora have been created through a set of scenarios. Dedicated machines have been set-up and the research group had to perform actions on these systems at particular times defined by scenarios. These actions can include the download of contents, sending e-mails to another system, browsing the Web or interact with documents. The whole process was mostly manual, even if some automation has been performed regarding the Web browsing. This solution enables the creation of corpora of

great quality. Despite this, the time required is significant. Besides this, if a user made a mistake without realising it, the whole process might be compromised.

NIST developed a range of disk images based on scenarios such as a hacking scenario or a mobile-based scenario. These images have been made publicly available only recently as part of the Computer Forensic Reference Data Sets (CFReDS) [43] project. Carrier [44] created a similar digital corpora repository, and proposes a list of disk images and memory dumps scenarios. While these scenarios could be useful for teaching and training purpose they have limitations for testing purpose. It is possible that no scenario exist which could be used to evaluate a particular function. These digital corpora cannot be used as a foundation for the creation of additional digital corpora as they are static. Hence, it is not possible to build similar digital corpora in order to confirm that the digital corpora do not introduce bias during the evaluation of the DFTs.

Additional work needs to be carry out in order to create additional data to create digital corpora which mimics real user activities. Research from other fields might be of interest to help researchers to create or improved existing digital corpora. However, most studies focus on user characteristics which do not have any link with the creation of digital corpora. These characteristics include genders [45] [46], ethnicity [47], or the rate of people which use computers at work [48] and at home [49].

6 Conclusion

This paper has outlined evaluation and validation methodologies, where some of these are too complex to be used by digital forensics investigators such as Carrier's abstraction layers model [19], and others do not cover all aspects of the tools [32]. For all them, none has been implemented in such a way that enable automations of the validation process. This means that testing may need to be performed manually. This is obviously an issue as it takes away a significant amount of time from investigators.

Beckett's [3] methodology can be used to define the requirements to validate digital forensics functions. This is a good methodology which covers all aspects in the definition of the validation process. However, the methodology does not cover the actual implementation of the validation process. Therefore, another methodology is needed. A good candidate is the methodology of Wilsdon [13] based on black-box testing.

7 Bibliography

[1] M. Reith, C. Carr, and G. Gunsch, "An examination of digital forensic models," *International Journal of Digital Evidence*, vol. 1, no. 3, pp. 1–10, 2002.

[2] R. Leigland and A. W. Krings, "A formalization of digital forensics," *International Journal of Digital Evidence*, vol. 3, no. 2, pp. 1–32, 2004.

[3] J. Beckett and J. Slay, "Digital forensics: Validation and verification in a dynamic work environment," in *Proceedings of the 40th Hawaii International Conference on System Sciences*, 2007.

[4] T. Wilsdon and J. Slay, "Digital forensics: exploring validation, verification & certification," in *First international workshop on Systematic Approaches to Digital Forensic Engineering*, (Taipei), 2005.

[5] J. Robertson, "The future of forensic science," in *Asia Pacific Police Technology Conference*, (Canberra, Australia), 1991.

[6] M. Meyers and M. Rogers, "Computer forensics: The need for standardization and certification," *International Journal of Digital Evidence*, vol. 3, no. 2, 2004.

[7] D. Gollman, *Computer Security*. Glasgow,: Bell & Bain, 2006.

[8] GIAC, "Giac certified forensic analyst (gcfa)," 2000.

[9] ISO/IEC, "Iso/iec 17025:2005 general requirements for the competence of testing and calibration laboratories," 2005.

[10] NIST, "Computer forensics tool testing (cftt) project overview," January 2011. 26 January 2011.

[11] DoD, "The defense cyber crime institute (dcci)," June 2010. [Retrieved 28 January 2012].

[12] L. Pan and L. M. Batten, "Robust performance testing for digital forensic tools," *Journal of Digital Investigation*, vol. 6, pp. 71–81, 2009.

[13] T. Wilsdon and J. Slay, "Validation of forensic computing software utilizing black box testing techniques," (Perth Western Australia,), 4th Australian Digital Forensics Conference, Edith Cowan University, 2006.

[14] NIST, "Test results for disk imaging tools: dd gnu fileutils 4.0.36, provided with red hat linux 7.1," August 2002. [Retrieved 28 January 2012].

[15] The Free Software Foundation, "http://linux.die.net/man/1/dd." die.net, 2010. [Retrieved 21 January 2012].

[16] J. Kornblum, "The linux kernel and the forensic acquisition of hard disks with an odd number of sectors," *International Journal of Digital Evidence*, vol. 3, no. 2, pp. 1–5, 2004.

[17] M. Pollitt, "Principles, practices, and procedures: an approach to standards in computer forensics," in *Second International Conference on Computer Evidence*, (Baltimore, Maryland, United States of America), 1995.

[18] M. Hildebrandt, S. Kiltz, and J. Dittmann, "A common scheme for evaluation of forensic software," in *Sixth International Conference on IT Security Incident Management and IT Forensics*, pp. 92–106, IEEE, 2011.

[19] B. Carrier, "Defining digital forensic examination and analysis tools," in *Digital Forensic Research Workshop 2002*, 2002.

[20] L. Pan and L. M. Batten, "Reproducibility of digital evidence in forensic," in *Digital Forensic Research Workshop (DFRWS)*, 2005.

[21] NIST, "Digital data acquisition tool specification," October 2004. [Retrieved 4 February 2012].

[22] NIST, "Digital data acquisition tool test assertions and test plan," November 2005. [Retrieved 4 February 2012].

[23] NIST, "Hardware write blocker device (hwb)." National Institute of Standards and Technology, October 2003. [Retrieved 22 October 2011].

[24] NIST, "Hardware write blocker (hwb) assertions and test plan," March 2005. [Retrieved 4 February 2012].

[25] NIST, "Forensic string searching tool requirements specification," January 2008. [Retrieved 4 February 2012].

[26] Y. Guo, J. Slay, and J. Beckett, "Validation and verification of computer forensic software tools - searching function," *Journal of Digital Investigation*, vol. 6, pp. 12–22, 2009.

[27] Y. Guo and J. Slay, "Data recovery function testing for digital forensic tools," in *Advances in Digital Forensics VI - Sixth IFIP WG 11.9 International Conference on Digital Forensics*, vol. 337, pp. 297–311, Springer, 2010.

[28] Y. Guo and J. Slay, "A function oriented methodology to validate and verify forensic copy function of digital forensic tools," in *International Conference on Availability, Reliability and Security*, 2010.

[29] J. Reust, "Dfrws 2005 workshop report," tech. rep., Digital Forensic Research Workshop, 2006.

[30] J. Pan, "Software testing," 1999. [Retrieved 11 January 2012].

[31] W. G. Kruse and J. G. Heiser, *Computer Forensics Incident Response Essentials*. Addison Wesley, 2002.

[32] B. Lempereur, M. Merabti, and Q. Shi, "Pypette: A framework for the automated evaluation of live digital forensic techniques," in *11th Annual PostGraduate Symposium on The Convergence of Telecommunications Networking and Broadcasting*, 2006.

[33] NIST/SEMATECH, "e-handbook of statistical methods." NIST/SEMATECH, 2006. [Retrieved 16 February 2012].

[34] G. Tagushi, *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Quality Resources, 1986.

[35] J. R. Lyle, "If error rate is such a simple concept, why don't i have one for my forensic tool yet?," *Journal of Digital Investigation*, vol. 7, pp. 135–139, 2010.

[36] The Supreme Court, "Daubert v. merrell dow pharmaceuticals," *New England Journal of Medicine*, vol. 509, pp. 585–589, 1993.

[37] W. Buchanan, "Advanced security and network forensics," 2011.

[38] D. Ayers, "A second generation computer forensic analysis system," *Journal of Digital Investigation*, vol. 6, pp. 34–42, 2009.

[39] L. Pan and L. Batten, "An effective and efficient testing methodology for correctness testing for file recovery tools," in *Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2007.

[40] S. Garfinkel, P. Farrell, V. Roussev, and G. Dinolt, "Bringing science to digital forensics with standardized forensic corpora," *Journal of Digital Investigation*, vol. 6, pp. 2–11, 2009.

[41] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classi cation research," *Machine Learning: ECML 2004*, vol. 1, pp. 217–226, 2004.

[42] K. Woods, C. A. Lee, S. Garfinkel, D. Dittrich, A. Russell, and K. Kearton, "Creating realistic corpora for security and forensic education," in *Naval Postgraduate School Monterey Ca Dept Of Computer Science*, 2011.

[43] NIST, "The cfreds project." National Institut, July 2011. [Retrieved 27 April 2012].

[44] B. Carrier, "Digital forensics tool testing images." SourceForge, 2010. [Retrieved 28 April 2012].

[45] P. Schumacher, "Gender, internet and computer attitudes and experiences," *Computers in Human Behavior*, vol. 17, no. 1, pp. 95–110, 2001.

[46] N. Lia and G. Kirkup, "Gender and cultural differences in internet use: A study of china and the uk," *Computers & Education*, vol. 48, no. 2, pp. 301–317, 2007.

[47] K. Korgen, P. Odell, and P. Schumacher, "Internet use among college students: Are there differences by race/ethnicity?," 2001. [Retrieved 15 March 2012].

[48] BLS, "Computer and internet use at work summary," August 2005. [Retrieved 15 March 2012].

[49] U.S. Census Bureau, "Computer and internet use," 2009. [Retrieved 15 March 2012].