Forensic Science International: Digital Investigation 52 (2025) 301874

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

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

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

A metrics-based look at disk images: Insights and applications

Lena L. Voigt [a,*], Felix Freiling [a], Christopher Hargreaves [b]

[a] Department of Computer Science, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany
[b] Department of Computer Science, University of Oxford, Oxford, UK

ARTICLE INFO

Keywords:
Forensic datasets
Dataset assessment
Synthetic data realism
Digital forensic education

ABSTRACT

There is currently no systematic method for evaluating digital forensic datasets. This makes it difficult to judge their suitability for specific use cases in digital forensic education and training. Additionally, there is limited comparability in the quality of synthetic datasets or the strengths and weaknesses of different data synthesis approaches. In this paper, we propose the concept of a quantitative, metrics-based assessment of forensic datasets as a first step toward a systematic evaluation approach. As a concrete implementation of this approach, we introduce *Mass Disk Processor*, a tool that automates the collection of metrics from large sets of disk images. It enables a privacy-preserving retrieval of high-level disk image characteristics, facilitating the assessment of not only synthetic but also real-world disk images. We demonstrate two applications of our tool. First, we create a comprehensive datasheet for publicly available, scenario-based synthetic disk images. Second, we propose a formal definition of synthetic data realism that compares properties of synthetic data to properties of real-world data and present results from an examination of the realism of current scenario-based disk images.

1. Introduction

Datasets have been identified as fundamental to the development of digital forensics as a science (Garfinkel et al., 2009). There has been some effort to classify (Breitinger and Jotterand, 2023) and to index them (NIST, 2024a). However, cataloging specific, detailed properties of available datasets has not been performed, meaning that locating a dataset that is suitable for a specific purpose is still challenging.

Synthetic disk images have become an essential part of the available data since the release of real-world data is highly restricted for security, privacy, and legal reasons. Even sharing synthetic data can be difficult if care is not taken, as it may become contaminated with information from the creator or creation environment (e.g., IP addresses). While scrubbing data is an option, it can be challenging to be comprehensive and maintain a consistent image.

Nevertheless, synthetic disk images remain the best alternative to real-world data and are even preferable in some use cases. For example, in tool testing, synthetic disk images might focus on edge cases or unlikely scenarios to test the capabilities of tools under challenging or atypical conditions (i.e. error-focused datasets) (NIST, 2024b; Hargreaves et al., 2024b). In other contexts, such as scenario-based teaching, efforts have been made to automate parts of the disk image synthesis process and to make synthetic disk images better resemble those encountered in real-world investigations (Moch and Freiling, 2009, 2012; Du et al., 2021; Göbel et al., 2022; Schmidt et al., 2023; Wolf et al., 2024; Voigt et al., 2024). Others have questioned whether realism is necessary (Göbel et al., 2023), or have considered what realism means in the context of synthetic disk images (Voigt et al., 2024).

Individual efforts to improve the scenario-based data generation process have suggested that certain metrics might be used to assess the benefits of different methods (Du et al., 2021; Voigt et al., 2024). However, a general approach for evaluating one automation framework against another has not been explored, and certainly not operationalized.

Through the design, implementation, and application of a system to collect a series of metrics from disk images at scale, this work makes several contributions in these areas:

- Design and development of a software framework to extract, organize, and present metrics for large sets of disk images.
- Collection and indexing of 25 publicly available, sce-nario-based disk images, with a demonstration of the framework used to collect metrics from them.
- Creation of datasheets summarizing 98 metrics for each of the 25 publicly available, scenario-based disk images.

* Corresponding author.
E-mail addresses: lena.lucia.voigt@fau.de (L.L. Voigt), felix.freiling@fau.de (F. Freiling), christopher.hargreaves@cs.ox.ac.uk (C. Hargreaves).

https://doi.org/10.1016/j.fsidi.2025.301874

Available online 24 March 2025

- Proposal of a formalized definition of realism in the context of synthetic disk images.
- Demonstration of applying the proposed definition using comparative metrics from 11 real-world disk images, the 25 publicly available disk images, and 37 internal disk images created for scenario-based digital forensics teaching in university environments.

The remainder of this paper is structured as follows: Section 2 provides background and related work in this area. Section 3 explains the methodology used in this work. The results are presented in the following sections: Section 4 describes the software framework used to extract disk metrics. Section 5 details the results of the data collection and analysis of publicly available, scenario-based disk images. Section 6 provides a definition of realism based on a security game-inspired model, followed by insights gained from this quantitative view of realism in Section 7, where synthetic disk images are compared with those containing 'real-world usage.' Finally, Section 8 discusses the overall results and outlines future work, and Section 9 provides the conclusion. A summary table for the metrics retrieved for the public datasets, individual datasheets, and the code for our metrics-retrieval framework are available in a repository.[1]

2. Related work

Some of the previous work on using automation to enhance synthetic disk images that was discussed in the introduction has utilized specific metrics as a means of inspecting the results of disk image synthesis. Du et al. (2021) used *Plaso* on synthesized disk images to examine the increase in events after automated actions were performed on a system. The timeline entry types considered were for files of the following types: event log, Windows Registry, web history, log, PE, link, and OLECF. Schmidt et al. (2023) collected a similar set of metrics with Plaso, focusing on events related to event logs, Windows Registry hives, SQLite databases, OLECF, prefetch, and link files. They compared the metrics of their proposed computer vision approach to Powershell and Python-based user emulation approaches. More recently, Voigt et al. (2024) included metrics from different sources, including the number of files listed by *fiwalk*, the number of days between the first and last file system creation time, and the total events listed by *log2timeline*.

Commercial tools also incorporate metrics to some extent. For example, Magnet AXIOM presents the 'Number of artifacts' in the case overview after extraction, dividing them by category into: operating system, media, web-related, documents, refined results, custom, application usage, communication, encryption & credentials, email &calendar, and cloud storage.

Overall, metrics have been considered in digital forensics, either to provide an overview of potentially interesting artifacts in data sources added to a case or to measure the success of automation frameworks in improving synthetic disk images. However, most prior work has relied on Plaso or fiwalk output, meaning the results were limited to the number of events (of different categories), counting files, or the difference between file system timestamps. None have developed a bespoke set of metrics, systematized them around specific desirable attributes for synthetic disk images, or have provided a mechanism to calculate those metrics at scale.

3. Methodology

The limitations of prior work described above lead to the guiding question of this paper: Can a metrics-based view help us assess the status quo of publicly available, scenario-based synthetic disk images and assess their level of realism? In the following, we describe the methodology employed to address this question.

3.1. Data collection

To compile our dataset, we collected two distinct types of disk images: synthetic disk images and real-world disk images, only considering scenario-based disks for the former.

To be able to commence our data collection, we first had to clarify what is meant by *real-world*, *synthetic*, and *scenario-based* disk images. For our data collection, leaning on existing definitions of Breitinger and Jotterand (2023) and Garfinkel et al. (2009), we define a *real-world disk image* as follows:

Definition 1. **(Real-world disk image)** *A* real-world disk image *is an image of a disk on which regular day-to-day activities were carried out by one or more users without the intention of creating data for digital forensic analysis or investigation.*

This contrasts with our definition of *synthetic* disk images. Those disk images were produced with the intention of creating data that can be utilized for digital forensic purposes.

Our use of the term *scenario-based* data corresponds to the definition of Breitinger and Jotterand (2023): "Scenario data, on the other hand, has a higher complexity, is generated over a longer time frame, or is based on real scenarios (as it tries to mimic them)." Therefore, we considered synthetic disk images created in accordance with a scenario for digital forensic investigation, such as disks from the M−57 scenarios in the Digital Corpora (Garfinkel, 2024) or disk images created for forensic capture-the-flags (CTFs) with a storyline.

With this understanding of the terminology, we subsequently compiled our three datasets of publicly available scenario-based, internal scenario-based, and real-world disk images. For simplicity, in the following, we refer to them as *Public*, *Internal*, and *Real-world* datasets or disk images, respectively.

We conducted our data collection in September 2024, focusing exclusively on disk images originating from Windows systems. We excluded other operating systems and storage media without an operating system, such as USB drives. Our Public dataset is a best-effort collection of publicly available, scenario-based disk images, see Appendix A. We searched the Digital Corpora (Garfinkel, 2024) and CFReDS (NIST, 2024a) repositories, using a snowballing approach to find further data, and conducted Google searches for forensic scenario disk images. The dataset includes disk images published by digital forensics educators or companies, teaching scenarios, or scenario-based CTFs. We excluded disk images that were only available for download with considerably limited transfer rates.

Additionally, we compiled an Internal dataset, acknowledging that while many educational institutions do not publicly release their synthetic disk images, they still provide valuable insight into the data used for teaching. We collected scenario-based synthetic Windows disk images employed for educational and demonstration purposes from five different institutions.

Ultimately, we acquired a limited set of Real-world disk images available to the researchers. This dataset comprises drives from personal computers that were used without the intention of generating data for this purpose. These drives were in use by the researchers between June 2012 and September 2024. It is worth noting that one of the drives originated from a virtual machine, yet still fits our definition of real-world usage.

3.2. Collecting a comprehensive set of disk image metrics

To enable the retrieval of metrics from disk image datasets, we developed the extensible Mass Disk Processor (MDP) framework. This framework includes plugins for collecting various metrics and summarizing the results for all disks in a dataset. We give details on the MDP framework and the types of metrics collected in Section 4.

[1] https://github.com/lenavoigt/mass-disk-processor.

3.3. Analysis and presentation of results

Finally, we analyzed the results from the MDP metric retrieval for the Public, Internal, and Real-world datasets. We inspected values across different categories and datasets, calculating the mean, median, range, variance, and standard deviation. Upon examining the metrics, we observed that most metrics did not appear to be normally distributed. Therefore, we predominantly use the median to report our findings after introducing the MDP framework.

4. The Mass Disk Processor (MDP) framework

In this section, we describe the system design and currently supported plugins of the Mass Disk Processor (MDP) framework, which we implemented for bulk metrics retrieval of disk image datasets. Moreover, we outline the application of MDP for extracting metrics from our datasets, detailing the employed configurations and parameters.

4.1. System design

The Mass Disk Processor (MDP) framework is built on *pytsk* (Cohen, 2024) and *libewf* (Metz, 2024) to provide access to disk image partitions and file systems. It then provides a wrapper around those packages, combined with a plugin-based architecture that enables the collection of data points or metrics from disk images. Individual plugins also leverage other Python libraries that offer relevant functionalities, such as *pyregistry* (Ballenthin, 2024b) and *python-evtx* (Ballenthin, 2024a). It is also possible to integrate existing tools like *Plaso*.

We have designed the MDP framework to allow for the activation and deactivation of plugins, supporting use-case-specific configurations, such as restricting plugins to ensure privacy-preserving metrics when collecting real-world data. For bulk processing of disks, users need to specify a primary directory containing subdirectories, each corresponding to different cases with one or more disk images. Additionally, users must select the plugins to be utilized for obtaining metrics from the disk image dataset.

The MDP framework offers optional preprocessing steps, such as retrieving file signatures and computing SHA-1 hashes for files below a specified size limit. While these steps can increase processing time considerably, they are necessary to run specific plugins, as signature mismatch counts and comparison of file hashes against a given hash set. Without appropriate preprocessing, these plugin outputs are omitted, and a warning is given. Additionally, there is an optional feature to store file lists, along with their hashes and signatures, in a database. This stored information can be leveraged in subsequent processing runs.

The bulk disk image processing in MDP generates a summary in both TSV and JSON format, providing an overview of the outputs from all plugins for each disk processed in a run. Within each case folder, plugin results are also stored separately, including a plugin description, the source file path, and a timestamp indicating when the result was generated.

4.2. Supported plugins

Within MDP, we have focused our implementation on Windows-based plugins, as Windows has maintained the largest global market share among desktop operating systems for personal computers since at least 2009, with a share of 73.4 % as of September 2024 (StatCounter, 2024b), mirroring the report of Windows being most commonly encountered by practitioners in the DFPulse 2024 Practitioner Survey (Hargreaves et al., 2024a). However, selected cross-platform modules are available. Appendix B gives an overview of the plugins currently implemented.

The plugins are categorized according to five types of high-level properties of disk images, which are called *Configuration*, *Longevity*, *Activity*, *Volume*, and *Notables*.

The *Configuration* plugins provide insight into the system setup by extracting details such as disk size, number of partitions, and the presence of various operating system types (Windows, macOS, Linux). For Windows systems, they can provide additional information about the Windows version, user and application count, screen resolution, and the presence of different browsers (Chrome, Edge, Firefox) along with the default browser.

The available *Longevity* plugins retrieve the time span of both the file system and the operating system, providing insight into the duration of system usage. The current file system time span plugin analyzes the creation timestamps of files. For Windows, the lifetime plugin extracts Windows Registry data to determine the installation date and the last shutdown time.

The *Activity* plugins in MDP gather several indicators of system use and user interaction. They collect data from Windows event logs on the overall number of logins, failed and successful login attempts, and unlock events. Additionally, the plugins retrieve the total number of link files and recent link files in user folders and the start menu, the count of prefetch files, and the number of USB mass storage connections. For the browsers Chrome, Edge (v79+), and Firefox, the plugins record the overall counts of website visits and the number of searches across the search engines Google, Bing, and DuckDuckGo from the browser history.

The *Volume* plugins in MDP focus on measuring the quantity of files on the disk image. They provide counts of the total number of files and the number and percentage of files in user directories. Files are categorized and counted by type based on typical file extensions, e.g., for audio, compressed, office, PDF, and video files. Additionally, there are plugins to calculate total disk usage and file size metrics, including mean and median file sizes. It is also possible to count the number of files that are not included in a hash database of known files, such as Reference Data Sets (RDS) of the National Software Reference Library (NSRL) (NIST, 2024c). For counting non-NSRL files, an NSRL RDS comprising known SHA-1 hashes must be specified to compare with file hashes on the disk image. It should be noted that for NSRL databases, this often means that the count of non-NSRL files also excludes zero-sized files.

The *Notables* plugins in MDP are designed to identify potential anomalies or points of interest. Currently, they count files with mismatches between their signature and extension, as well as the number of clock change events. The latter is also included in the Activity category. Additional plugins, for instance, for identifying encrypted files (such as VeraCrypt containers or encrypted ZIP files), detecting specific keywords, or locating files present in a hash database of notable files, are conceivable for future development. However, the focus has so far primarily been on plugins from the other categories.

4.3. Dataset processing setup

To process the Public, Internal, and Real-world datasets considered in this work, we utilized all available MDP plugins except the Plaso plugin. Although Plaso's integration was initially demonstrated, our current analysis relies solely on custom MDP plugins. These plugins provide precise control over the origins of metrics retrieval and help ensure privacy-conscious handling of real-world data.

For the non-NSRL file count, we used the NSRL RDS 2024.*03.1 Modern Minimal*. During preprocessing, we retrieved the first eight bytes of each file on a disk image to check the file signatures and computed SHA-1 hashes for all files with sizes up to the maximum file size in the selected NSRL RDS (approximately 137.5 GB). Furthermore, we chose not to store file lists with hashes and signatures, given that we work with real-world data.

5. Insights from Public disk image metrics

Selecting an appropriate disk image for specific purposes, for example, in digital forensic education, requires an understanding of the characteristics of available synthetic disk images. Despite the limited

number of public disk images from forensic dataset collections, company websites, or individual sources, manually assessing the suitability of each image is a time-consuming task. Although forensic datasets often provide descriptions, metadata records, or tags, they may omit features of particular interest to educators. Solution notes accompanying some public disk images can aid in understanding the disk image's properties. However, forensic data publishers may choose to make solutions available only upon request or not at all. Moreover, the need to navigate solution write-ups to get a first impression of an image's suitability poses a barrier to data reuse.

To address this challenge, we propose to utilize MDP to retrieve metrics about disk image collections automatically. With this, users can identify the most relevant disk images for the purpose they have in mind before having to conduct a further in-depth, manual analysis. Additionally, MDP-retrieved metrics offer a means of assessing the current state of publicly available, scenario-based synthetic disk images.

In this section, we present selected insights from our assessment of publicly available, scenario-based disk images. We collected a total of 25 disk images from different resources (see Appendix A) that meet the selection criteria for scenario-based synthetic Windows disk images we detail in Section 3. We created an overview of their properties retrieved with MDP as well as a datasheet for each of them that can be found in our repository. Table 2 shows selected metrics from different categories that we now discuss in more detail.

5.1. Configuration

Within our set of Public disk images, Windows 10 is the most represented, with eight images. Windows XP follows with six images. Windows 11 and Windows 8.1 are each found in three images, while Windows 7 appears in two. Additionally, our collection includes one disk image each of Windows Server (2008), Windows Server 2022, and Windows Vista (see Table 1).

The median of installed applications listed in the uninstall Registry key is 29 in our Public dataset, with a maximum of 309 entries in the Windows 7 disk image MagnetCTF20. Additionally, the Edge browser is present on all disk images, while Chrome is found on 52 %, and Firefox is present on 44 % of the disk images.

5.2. Longevity

To determine the lifetime of the systems associated with the disk images, we retrieved two different types of metrics: the file system time span and the Windows operating system lifetime. We observed that the file system time span we retrieved from the *created* timestamps gave little insight into the actual lifetime of the system due to files associated with software on the system having timestamps earlier than the system was set up. An alternative metric for assessing the file system time span could involve retrieving accessed or modified timestamps, which may provide more insights. This metric could be considered alongside the Windows lifetime, which we solely focus on in the following. It should be noted that scenarios might include the tampering of values we retrieve for our metrics, highlighting the goal of establishing several metrics for the same aspect.

The first setup system in the dataset is NIST04, a Windows XP installation from August 2004. The newest system is Bart23, featuring Windows 11 installed in October 2023. The median Windows lifetime is 23 days. The majority of systems, specifically 19 out of 25, were used for less than three months. Only one system, the Windows 10 DFRWSRodeo24, has a calculated lifetime exceeding one year. This system also accounts for the most recent shutdown recorded in our dataset, which occurred in March 2024.

5.3. Activity

The total login count across systems shows a median of 11 logins, with the highest count of 96 logins observed in the M57-09Pat system. Regarding browser history, there is a median of 236 website visits and 15 searches. The richest browser history is found in LoneWolf18, with 2,289 website visits and 229 searches.

5.4. Volume

For the disk images in our Public dataset, the median total number of files is 125.6K, though this figure alone is not particularly illustrative. To provide additional context, the number of files in user folders has a median of 6.2K, with the MagnetCTF23 system having the highest number at 111.2K. Furthermore, the number of non-NSRL files has a median of 77.0K, with OpenUni22 having the most at 231.1K.

6. A quantitative view on realistic disk images

From our collection process, we know that the disk images from the previous section were synthetic. Had we not known this, some of the statistics (especially from the Longevity category) might strongly hint towards categorizing them as synthetic. However, it remains highly unclear whether scenario-based disk images can be clearly distinguished from real-world disk images. This observation is relevant in the context of discussions in the literature about how 'realistic' scenario-based disk images are or need to be (Voigt et al., 2024). While realism is a desirable property often intuitively postulated by early disk image synthesis tools (Moch and Freiling, 2009), it is a rather evasive concept that is hard to define precisely.

Realism should certainly not be considered the sole quality criterion. For example, disk images containing edge cases like partition loops or ambiguous file system partitions (Schneider et al., 2022) and other error-focused datasets (Hargreaves et al., 2024b) may be less likely to be encountered in practice but are still vital for testing the limits of analysts and tools. Nevertheless, a metrics-based look at disk images can help to establish a definition of realism that is both precise and useful.

6.1. Definition based on indistinguishability

There are many possible ways to define what we mean by a synthetic disk image being 'realistic.' Intuitively, a synthetic image can be considered realistic if it 'looks similar' to a real-world disk image as defined in Definition 1.

We use this intuition to define the concept of *realism of synthetic disk images* using tools borrowed from cryptography where central security definitions are based on the concept of indistinguishability: Roughly speaking, if an attacker cannot distinguish an encrypted message from a random string, then the encryption technique guarantees confidentiality in the given circumstances. Since there are many attack strategies, such security definitions are based on the notion of a *game* between an attacker and defender that behave according to predefined rules. The

Table 1
Number of *Public* and *Internal* scenario-based synthetic disk images as well as *Real-world* disk images in our dataset.

Windows Version	Public	Internal	Real-world
Windows 11	3	–	2
Windows 10	8	11	5
Windows 8.1	3	–	–
Windows 7	2	19	3
Windows Vista	1	1	–
Windows XP	6	5	1
Windows Server 2008	1	–	–
Windows Server 2022	1	–	–
Windows Server 2019	–	1	–
Total	**25**	**37**	**11**

Table 2
Summary of a subset of 98 metrics computed with MDP for the 25 publicly available, scenario-based disk images listed in Table 3.

	Configuration		Longevity	Activity			Volume		
ID	**Version**	**Applications**	**Installation/Shutdown**	**Logins**	**Browser Visits**	**Browser Searches**	**Files**	**Non-NSRL Files**	**Files in User Folder**
Bart23	Windows 11	16	2023-10/2023-10	10	278	50	257394	97824	13780
BelkaCTF1	Windows 10	23	2020-08/2021-02	6	80	15	154706	86277	13443
BelkaCTF5	Windows 10	24	2022-06/2022-07	0	464	87	386144	85805	23859
CCIKip	Windows 7	114	2014-01/2014-01	14	82	9	68304	20864	4151
CCITucker	Windows 8.1	17	2013-12/2013-12	4	156	10	89883	15222	3656
CellebriteCTF21	Windows 10	44	2021-03/2021-07	0	154	30	410790	145014	48299
DefenitCTF20	Windows 10	24	2020-05/2020-05	0	70	24	125575	81481	5705
DFRWSRodeo24	Windows 10	20	2019-03/2024-03	33	0	0	267949	139864	4916
Hadi1	Win. Server 08	13	2015-08/2015-09	0	0	0	55671	19844	219
Hadi2	Windows 8.1	48	2016-06/2016-06	3	421	150	114137	16499	11224
Hadi3	Windows 8.1	14	2015-12/2015-12	3	0	0	91879	7955	1924
InCTF20	Windows XP	39	2020-03/2020-07	22	85	12	12899	6047	2710
LoneWolf18	Windows 10	29	2018-03/2018-03	23	2289	229	152543	104577	8364
M57-08	Windows XP	40	2008-05/2008-07	84	489	40	31909	17834	6226
M57-09Charlie	Windows XP	113	2009-11/2009-12	51	1080	40	29555	14598	4475
M57-09Jo	Windows XP	108	2009-11/2009-12	32	422	17	31073	15799	4958
M57-09Pat	Windows XP	111	2009-11/2009-12	96	295	13	36765	20763	11989
M57-09Terry	Windows Vista	38	2009-11/2009-12	24	140	0	82988	39185	6140
MagnetCTF19	Windows 10	13	2018-07/2019-03		592	2	162283	111612	8103
MagnetCTF20	Windows 7	309	2020-02/2020-04	24	471	115	162533	76977	10164
MagnetCTF22	Windows 11	16	2022-02/2022-02	7	236	14	324089	161010	17596
MagnetCTF23	Windows 11	21	2022-11/2023-01	0	1078	113	253443	81328	111193
NIST_04	Windows XP	32	2004-08/2004-08	15	0	0	11501	7387	1197
OpenUni22	Win. Server 22	14	2023-09/2024-02		40	10	316515	231130	4643
Owl19	Windows 10	52	2017-01/2017-01	11	792	116	320849	104225	11409

defender uses a security primitive, like an encryption scheme, which the attacker attempts to break. If the defender consistently wins the game, the security primitive is considered secure.

Borrowing from standard cryptographic terminology of authentication protocols (Menezes et al., 1996, Chapter 10), we abstractly define two roles: the *prover* and the *verifier*. The prover has constructed a set of synthetic disk images and claims that they are realistic in the sense that they are indistinguishable from real-world disk images. The verifier is an entity that claims to be able to distinguish real-world from synthetic disk images. The game now consists of several rounds in which the verifier can study a given disk image and must eventually decide whether it is a real-world disk image or a synthetic one. Given multiple rounds of the game, the verifier may correctly classify some disks as synthetic but may also confuse synthetic ones with real-world ones. The prover wins if the best strategy that the verifier applies to distinguish synthetic from real-world disks is equivalent to random guessing; otherwise, the verifier wins. If the prover wins, the synthetic disk images produced by the prover can be considered realistic.

The definition implicitly assumes that both the prover and the verifier want to win the game and that the verifier has some common knowledge of what real-world disk images look like. The outcome of the game also depends on the type and amount of information that the verifier can derive from the image under consideration. In typical exercise situations, some aspects of the disk image are often 'out of scope' for the analysis, e.g., whether the image exhibits virtualization artifacts. Therefore, in our definition of realism, we want to regulate this information. Otherwise, the verifier is unrestricted in its activities; in particular, it can apply arbitrary computational resources.

6.2. The realism game

We begin by defining two sets of disk images, the set R of *real-world* images and the set S of *synthentic* images. Elements of S were artificially constructed by the prover, and elements of R were collected in the real world (as described in Section 3). To regulate the amount of information derivable from an image, we define a set $F = \{f_1, f_2, \ldots\}$ of *features* of disk images, where a feature is the result of computing a metric, i.e., a measurable value that can be derived from any image through a well-defined process. Examples of such features have been discussed above in Section 5. To restrict the verifier's queries, we define a subset $A \subset F$ of *allowed features*. This is useful when, in practice, certain aspects of the disk image should be ignored (e.g., virtualization artifacts).

The setting of the realism game is depicted in Fig. 1: We assume that the verifier has some world knowledge of real-world disk images that it can analyze arbitrarily to 'learn' as many aspects of realism as possible. The verifier then sits in front of a black box that hides the execution details of the game. Within the black box is also a (sufficiently large) reference set R of real-world disk images. The prover provides a set S of synthetic images.

Definition 2. **(Realism game)** *The* realism game *between prover and verifier operates in several rounds as follows: At the beginning of each round, one disk image D is randomly selected from either S or R. Then, the verifier can direct an arbitrary but finite number of queries for allowed features to the black box. If the requested feature f is allowed (i.e., if $f \in A$), then the black box returns the value of the measurement of f on D. Otherwise, the value $\perp$ ('undefined') is returned. Eventually, the verifier has to output its verdict of whether the image is a synthetic or real-world one. If the answer is correct (i.e., the image is correctly classified as coming from S or R) then the verifier wins. Otherwise, the prover wins. This ends the round, and the next round starts.*

6.3. Defining realism

This realism game is played for many rounds. In each round, a new disk image D is randomly selected. After n rounds, we calculate the probability of a correct choice as the number of wins of the verifier over n. This is a number between 0 and 1. If the verifier can reliably distinguish between synthetic and real-world images, then the number should be close to 1. Similarly, if the verifier consistently thinks a synthetic image is a real-world one and vice versa, i.e., if the verifier consistently loses, the number should be close to 0. If the verifier merely performs random guessing, the probability of winning is $\frac{1}{2}$. If the best that the

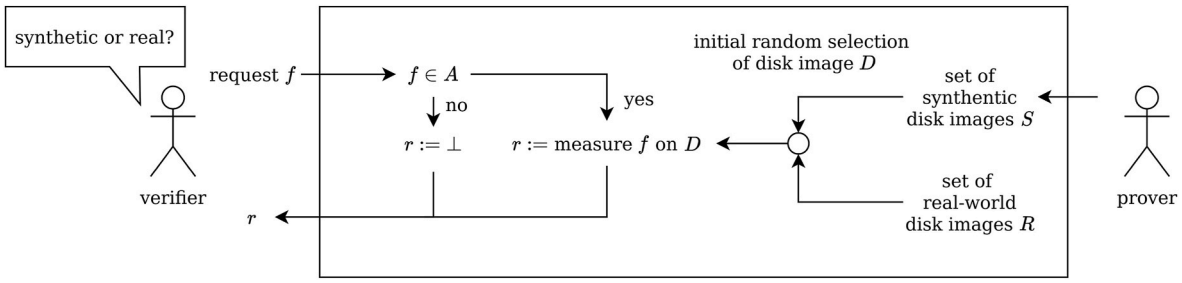

Fig. 1. The realism game: A verifier has to distinguish whether a disk image provided by a prover is a real-world or synthetic one. The image is selected at random. The verifier can make a series of feature queries and must eventually give a verdict of whether the image is a synthetic or real-world one.

verifier can do is not better than random guessing, the verifier cannot reliably distinguish real-world and synthetic disk images. And, if synthetic disk images cannot be distinguished from real-world disk images, we call the synthetic disk images realistic.

Definition 3. **(Realism of synthetic disk images)** *Let ω be the number of wins of the verifier within a sequence of n rounds ($0 \leq \omega \leq n$) of the realism game. We define the* advantage of the verifier after n rounds *of the realism game as follows:*

$$\alpha = |\frac{\omega}{n} - \frac{1}{2}|$$

If, for increasing numbers of n, the value of α approaches 0 (or generally is below some small number ϵ), we call the set S of synthetic disk images realistic with respect to the allowed feature set A.

7. Insights from a comparison of Synthetic and Real-world disk images

From our experience, the definition of realism from the previous section resembles how prior work has envisioned to approach the construction of 'interesting' disk images by successively expanding the generation capabilities of the synthesis tool to cover "most aspects of digital investigations" (Moch and Freiling, 2009, p. 80). Therefore, the degree of realism can be practically assessed by looking at selected properties of synthetic disk images and comparing them to those of real-world disks.

In the following, using the results of bulk processing our datasets with MDP, we illustrate selected findings from our metrics-based assessment of the scenario-based synthetic disk images, Public and Internal, compared to the Real-world disk images in our dataset. While we are aware that these results are not representative, they give first insights into the disparity of properties in scenario-based synthetic versus real-world datasets.

For certain metrics, we observed considerable differences across Windows versions within all three of our datasets. That is why we focus on presenting these metrics for Windows 10 disk images, as the overall dataset predominantly consists of Windows 7 and 10 images, with Windows 10 being more evenly distributed across the Public, Internal, and Real-world datasets. For brevity, we focus on the metrics from four categories: Configuration, Longevity, Activity, and Volume.

7.1. Configuration

When examining the Configuration category, three plugins showed notable results: the number of installed applications, the number of systems with different browsers (Edge, Chrome, Firefox) present, and the screen ratio.

Browsers Present: To assess the presence of browsers on the disk images, we can compare our datasets against browser usage statistics to determine how closely they reflect real-world trends. According to StatCounter (StatCounter, 2024a), as of September 2024, Chrome leads the global desktop browser market with a 64.8 % share, followed by Edge at 13.8 %, Safari at 9.2 %, and Firefox at 6.6 %. Although this statistic does not account for different operating systems, and we would not expect to find Safari on 9.2 % of our Windows machines, it highlights Chrome's prevalence and Firefox's relatively low usage. This trend dates back to January 2015, with Chrome consistently above 50 % and Firefox below 20 %.

In our dataset, Edge was unsurprisingly present on all Windows Systems. However, in all three datasets, Firefox's presence was higher than the usage statistic suggests: It appeared on 47.8 % of Public, 63.6 % of Real-world, and 38.9 % of Internal disk images, with the Internal dataset being closest to the usage statistics. This also underscores the limited representativeness of our Real-world dataset, showing a clear bias toward Firefox. Chrome was present in 56.5 % of Public, 17.8 % of Internal, and 81.8 % of Real-world systems. These shares for the Public and Real-world datasets align more closely with the market statistics, with Chrome notably underrepresented in the Internal dataset. It should be noted that we excluded the Windows server disk images from this comparison.

Screen Ratio: For the comparison of screen ratios, we focused on systems with Windows 8 and newer due to the compatibility constraints of our MDP plugin, and we further excluded Server versions. In modern computers, we would expect a considerable portion of systems to have a screen aspect ratio of 16:9 or 16:10. In our Real-world dataset, this expectation is met, with five out of seven systems having a 16:9 ratio and one system having a 16:10 ratio. However, in the Public dataset, only five out of 14 systems featured a 16:9 ratio, and none had 16:10. The Internal dataset showed no systems with either of these ratios. Instead, eight out of 14 Public and seven out of 11 Internal disk images had a 4:3 aspect ratio. This likely results from synthetic disk images often being created in virtual machines.

Installed Applications: The inspection of the number of installed applications retrieved from the uninstall Registry key revealed differences between the Windows 10 systems in our three datasets. In our Real-world dataset, the median number of applications listed per disk is 79.5, while the median number of applications in the Public and Internal datasets is 24 and 19, respectively.

7.2. Longevity

An examination of Windows operating system lifetimes across our Internal, Public, and Real-world datasets shows notable differences in system longevity, as illustrated in Fig. 2. In the Internal dataset, the median lifetime is 21.1 days, with a maximum of 338.4 days. The Public

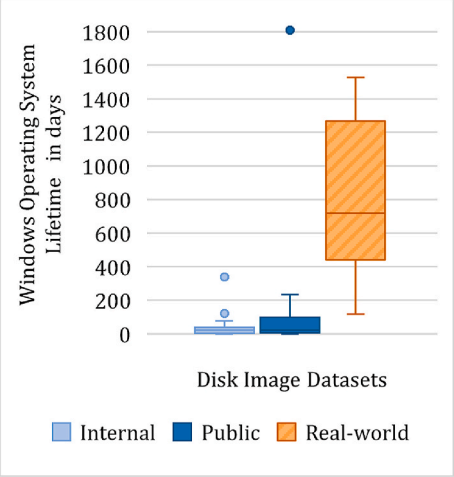

Fig. 2. Box plots and outliers for the Windows operating system lifetimes across our different datasets.

dataset shows a median lifetime of 22.8 days and a maximum of 1809.0 days, translating to almost 5 years. In contrast, the Real-world dataset presents a remarkably longer median lifetime of 720.2 days, approximately two years, with a maximum of 1526 days, or roughly 4.2 years.

7.3. Activity

Analyzing the Activity category reveals various differences in the value distributions across the datasets. We observed notable variations in the number of link files in the user folders in total, as well as the number of recent link files, the number of browser history visits and searches, the login count and the number of clock changes from the event logs, and the number of USB connections.

Link Files: Comparing the Windows 10 disk images from the different datasets, we observed the highest number of both total and recent link files in the user folders in the Real-world dataset, with medians of 321.5 and 144.5 files, respectively. In contrast, the Public dataset shows a median of 146 total and 22 recent link files. The Internal dataset displays the lowest median, with 130 total and 12 recent link files.

Browser History: The browser history metrics further differentiate the datasets, see Fig. 3. The Real-world dataset's median number of visits is 428, and the median number of searches is 156, with mean numbers at 3431.2 and 295.2, respectively. The Public dataset has a median of 236 visits and 15 searches, with means of 388.6 and 43.8. The Internal dataset again presents the lowest activity, with medians for both visits and searches at 0 and mean numbers of 53.7 visits and 8.9 searches. This is partially due to the fact that more than half of the disk images (21 of 37) in the Internal dataset have an empty browser history, compared to four of 25 in the Public dataset and only two of 11 in the Real-world dataset.

User Logins: For Windows 10 disk images, the Real-world dataset has a median login count of 619, with four out of five images exceeding 100 logins. The Public dataset's median is markedly lower, with only six logins. The Internal dataset shows a median of 17, with one image standing out with 10,145 logins. All other Internal and Public disk images in our datasets have login counts of less than 100.

Event Logs: We expected the number of clock change events to be an indicator of time manipulation attempts, for example, in forensic teaching scenarios. However, the analysis of the Windows 10 systems in our dataset revealed a median of 15 clock change events recorded in the event logs in our Real-world dataset, on systems where no time manipulation attempts had been made. In the Public and Internal datasets, the median number of such events was 8.5 and 4, respectively.

USB Connections: Differences are also apparent in the count of USB connections recorded in the SetupAPI.dev.log. The Real-world dataset exhibits the highest number of USB connections, with a median of 2.5 and a mean of 5.5. In comparison, the Public dataset's mean is at 1.1, while the Internal dataset's mean is at 1.3, with the medians being 0 and 1, respectively. Furthermore, the disk image with the highest number of USB connections in the Real-world dataset reaches up to 18 connections, compared to a maximum of eight in the Public and six in the Internal dataset.

7.4. Volume

There are also notable differences in the number of files stored on the systems between the synthetic datasets, Public and Internal, and the Real-world dataset. These differences are noticeable, e.g., in the total number of files, the files not included in the NSRL RDS, and the number of different file types. Since the number of files can vary considerably between different Windows versions, we present a comparison of the Windows 10 systems in the following. However, the general trend of disk images in our Real-world dataset containing a notably higher number of files was also observed across the other versions.

Number of Files: Real-world disk images exhibit the highest number of files per disk image, with a median of 547.1K files overall and 118.0K within user folders. In contrast, Public disk images have a median of 215.1K files overall and 9.9K within user folders. The Internal dataset shows the lowest medians, with 150.8K files and 4.0K in user folders.

Non-NSRL Files: Excluding files listed in the NSRL dataset reduces the

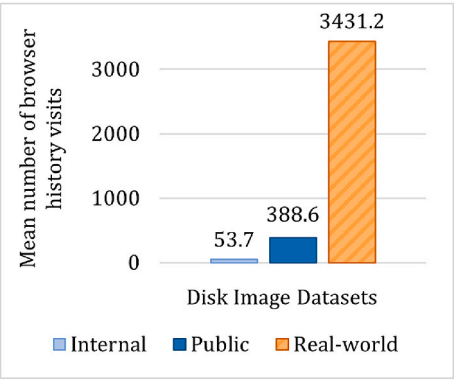

Fig. 3. Mean number of browser history visits across our different datasets.

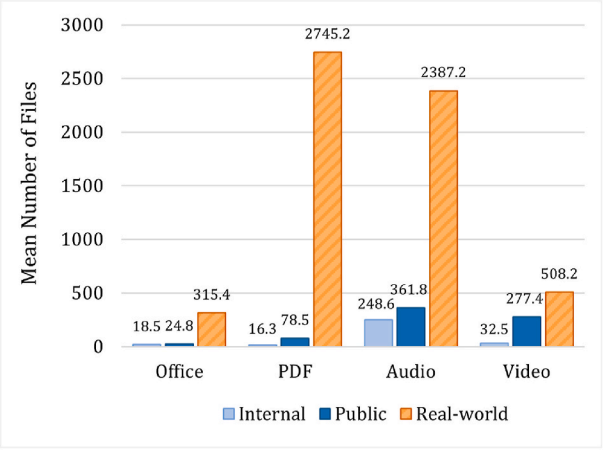

Fig. 4. Mean number of selected file types across our different datasets.

overall file count, yet a substantial number of files remains. The Real-world dataset shows a median of 405.0K files not in the NSRL. The Public dataset has a median of 104.4K non-NSRL files, while the Internal dataset has a median of only 52.8K.

File Types: This trend is also evident across various commonly user-generated file types, see Fig. 4. The mean number of office, PDF, audio, and video files in the Real-world dataset is notably higher than in the Internal and Public datasets.

8. Discussion

In this section, we discuss the need for both quantitative and qualitative metrics in assessing synthetic data realism, outline additional potential application scenarios for a metrics-based view on disk images, and describe limitations and future work.

8.1. Quantitative versus qualitative metrics

While quantitative metrics can provide many insights into the nature of scenario-based synthetic disk images, they have limitations in that they cannot capture certain properties that would be useful for assessment. Additional quantitative metrics could be derived, but not from the disk image itself, e.g., the number of relevant artifacts to locate within the disk image. However, there remain challenges in precisely grasping and expressing the notion of relevant evidence, which is dependent on the investigative hypotheses and the artifact's abstraction level (Gruber and Humml, 2023), e.g., whether a relevant artifact is a database file or a row within a database. The digital forensics tool *Autopsy* allows for the bookmarking of files and 'artifacts' (e.g., an extracted search term from a browser database), which could be used to describe these. Nevertheless, a more universal approach is needed, e.g., using the Cyber-investigation Analysis Standard Expression (CASE) (Casey et al., 2017).

However, other properties do not inherently lend themselves to quantitative metrics, e.g., the complexity of evidence recovery. Perhaps this could be proxied based on the artifact's location and assigning 'difficulty of interpretation' scores to the containing file type, but it is unclear how this would work in practice. Other qualitative measures are described by Voigt et al. (2024), e.g., "narrative coherence." This could also be combined with the measurement of scenario realism during the "scenario construction" and "storyboarding" stages (Hargreaves, 2017), e.g., whether the scenario captured by the disk image is feasible or probable in real-world investigations, likely needing the collaboration with practitioners for curriculum development called for in Hargreaves et al. (2024a).

These qualitative measures are important because it is possible to improve certain quantitative metrics without necessarily maintaining narrative coherence. For example, one could increase Volume metrics by depositing large numbers of random files onto the disk. This might be detectable through techniques that identify a large number of files being deposited during a single login session from an external source (Chow et al., 2007).

To enhance Activity scores, programs could be launched arbitrarily, or a browser could be set to visit random web pages. The latter could be detected if none of the visited web pages in the browser history are logically linked to one another (Hargreaves, 2009). Additionally, atypical browser usage, such as when users frequently access websites by manually typing complicated URLs, rather than navigating via clickable links, might be identified through transition types stored in Chrome's history (Boucher et al., 2022).

Moreover, Longevity could be improved through clock manipulations. However, this may lead to inconsistencies in cached content, which would not align with the supposed time of the actions being performed, potentially causing detectable discrepancies in timestamp or timing information (Vanini et al., 2024; Dreier et al., 2024).

Overall, content-aware, narrative-driven measures, qualitative or quantitative, are essential to provide not just a numerical high-realism but a disk image that is realistic in an intuitive sense. Nevertheless, the quantitative metrics used in this paper, as well as a framework for computing such metrics at scale, are critical first steps towards this next-generation realism.

8.2. Further application scenarios

While the benefits of a metrics-based look at disk images have been demonstrated for improving dataset accessibility and for defining the realism of scenario-based synthetic disk images, many other applications cannot be demonstrated in this paper due to space constraints and are, therefore, only briefly mentioned here.

Evaluating forensic artifact knowledge

When forensic knowledge of artifacts is applied at scale, in the course of developing MDP plugins, it is possible to identify limitations in that knowledge. For example, when recovering the Windows version, it became apparent that Windows 11 reports as Windows 10 in some Registry keys. This is documented online and likely known to forensic tool vendors, but no public information was found on this in a forensic context.

If those undertaking artifact research could express their findings in code and wrap them in an MDP plugin, then their 'artifact interpretation' hypothesis could be evaluated at scale.

Tool testing

Similar to testing artifact knowledge, the framework could also be used to test tools as scale. Command line tools at least, can be wrapped as an MDP plugin and applied to a large number of disk images to determine if the tool operates without crashing and that the output is as expected, which can be manually verified.

Triage

Disk metrics may also be applicable in the triage domain. Given some of the metrics calculated, e.g., Windows installation and shutdown, it could be possible to identify devices that were potentially in use during the time of an incident. Additionally, NSRL lookups could be substituted with a database of notable files. The number of link files, prefetch files, or other metrics within the *Activity* category could be used to identify a device of interest, similar to the "identification of 'hot drives"' suggested by Garfinkel (2006) and Patterson and Hargreaves (2012).

Forensic lab metrics

It may also be possible to correlate disk metrics with other case-related data, e.g., analysis time and outcome. By combining these inputs, insights could be gained for resource estimation for an unseen disk based solely on disk metrics calculated using a quick triage process.

Further insights into metric effectiveness

In the course of this research, it was determined that some metrics are less useful than initially expected. For example, the 'file system time span' did not provide as much insight as the 'Windows lifetime.' Although it was anticipated that the number of clock changes recorded in event logs would be higher on synthetic systems due to time manipulation, this was not the case. Therefore, when a metric is applied to large sets of disk images, it can be assessed whether the metric provides meaningful insights.

Metric-sharing for non-publishable disk images

Access to real-world datasets for forensic research remains severely restricted as it is "difficult to obtain and manage, and is increasingly surrounded by ethical and legal concerns" (Du et al., 2021). However, some research may not need full access to disk images, and high-level metrics may suffice. For example, software requirements for browser history visualization could be derived from the metrics related to browser history visits and searches, providing insights into the scale of

the visualization problem that needs to be addressed. The privacy implications of sharing high-level summary statistics are lower than those associated with the release of an individual's disk image.

8.3. Limitations and future work

The MDP framework currently has 21 plugins, generating 98 metrics. This is obviously a limited subset of all metrics that could be computed for a disk image. However, as MDP has been implemented with a plugin-based architecture, the addition of new plugins and, therefore, new metrics is straightforward. Furthermore, it is possible to run external tools via a plugin and capture and extrapolate output and metrics. Thus, not every metric needs implementation from scratch. While not presented in this paper, a Plaso wrapper is provided in our repository to allow re-implementation of the metrics of Du et al. (2021) and Schmidt et al. (2023).

However, the metrics selected have been designed to be minimally intrusive from a privacy perspective, e.g., by capturing only the total number of web searches or visits rather than details of their content. Consequently, future work could involve a study of volunteers' disks to collect additional real-world metrics for better comparison. Using tools like Plaso for this purpose is inappropriate for reasons of time and privacy.

There is a need for a larger-scale collection of real-world metrics to allow better comparison, as this research could only use 11 real-world disk images. Further work in this area could involve the volunteer data-donation study described earlier, with appropriate ethics and privacy procedures in place. It could also be complemented by an organization using MDP internally, where high-level metrics, even aggregate ones, could be collected. The summary statistics could be released with minimal privacy or legal concerns.

Another class of metrics that has not been implemented is cross-plugin metrics. For example, given the lifetime of a device and the number of logins, a normalized metric of 'mean logins per day' could be calculated. There are many other possibilities for more sophisticated metrics such as 'mean time between Google searches,' 'mean login duration' (combining login and logout events), or 'non-NSRL files in home folders,' which may further highlight differences between real-world disk images and synthetic ones.

Moreover, now that these metrics can be calculated, it is possible to assess the output from different automation frameworks. However, no publicly available disk images were found that were created with *TraceGen* (Du et al., 2021), *ForTrace* (Göbel et al., 2022), *pyautoqemu* (Schmidt et al., 2023), or *ForTrace++* (Wolf et al., 2024). Two were found for *Re-imagen* (Voigt et al., 2024), but they were not full scenario-based disk images. In future developments in this area, releasing datasets that demonstrate the capabilities of new or updated data synthesis frameworks would be beneficial, so that metrics can be calculated and compared. The open-source MDP framework can be used as the basis for that.

During our collection of scenario-based disk images, it became apparent that many scenario-based datasets are not disk images but mobile phone extractions, either backup-based or logical file extractions in ZIP or TAR format. This reflects the growing prevalence of mobile devices and, therefore, mobile device-based evidence. The MDP framework will soon be updated to process such datasets. As shown in Table 4 in the Appendix, some of the currently supported plugins are cross-platform, e.g., *chrome_history_entries*, *firefox_searches*, but others need updating to accommodate formats that are not disk-image-based, e.g., *no_files*. Additionally, new plugins need to be created for Android and iOS, e.g., *safari_history_entries*.

9. Conclusion

Availability of datasets is fundamental to the development of any science, and digital forensics has struggled with this for many years. Efforts have been made to collate datasets (Garfinkel et al., 2009; NIST, 2024a), categorize them (Breitinger and Jotterand, 2023), and to increase the quality of synthetic ones (Moch and Freiling, 2009, 2012; Du et al., 2021; Göbel et al., 2022; Schmidt et al., 2023; Wolf et al., 2024; Voigt et al., 2024).

This work has made multiple improvements in this area. It has indexed 25 publicly available, scenario-based disk images and developed an open-source framework that has enabled 98 different metrics to be computed for each of those disk images, resulting in an easy-to-access summary as well as detailed datasheets. The developed framework also has many other potential applications (discussed in Section 8.2).

Additionally, this work has addressed the ongoing lack of clarity regarding the *realism* of synthetic disk images. It has shown that using a cryptography-inspired 'realism game,' quantitative metrics can be used to measure this otherwise elusive concept. This has been demonstrated by comparing 11 real-world disk images against scenario-based synthetic ones, providing insights into the differences (based on Configuration, Longevity, Activity, and Volume) and, in some cases, potential deficiencies of data synthesis efforts.

However, this paper has also discussed the limits of a quantitative view of realism. Nevertheless, together with recent work by Voigt et al. (2024), it paves the way for a mixed approach to considering realism, where the quantitative aspects of *Configuration*, *Longevity*, *Activity*, and *Volume* could be combined with the qualitative aspects of *coherence* and *narrative*, that are described in Voigt et al. (2024). Combining these elements may enhance the quality of scenario-based synthetic disk images, offering potential benefits for digital forensic education, training, and beyond.

CRediT authorship contribution statement

Lena L. Voigt: Conceptualization, Methodology, Software, Investigation, Data Curation, Validation, Writing - Original Draft, Writing - Review & Editing Felix Freiling: Conceptualization, Methodology, Investigation, Writing - Review & Editing, Supervision, Funding acquisition Christopher Hargreaves: Conceptualization, Methodology, Software, Investigation, Writing - Review & Editing, Supervision.

Acknowledgments

Thanks to Paul Rösler for support with the definition of realism of synthetic disk images and to Katharina De Rentiis for assistance with collecting publicly available, scenario-based disk images. We thank the anonymous reviewers for their helpful comments. This work has been supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the Research and Training Group 2475 "Cybercrime and Forensic Computing" (grant number 393 541 319/GRK2475/2-2024).

Appendix

A. Public scenario disk images considered

Table 3 gives details of all public scenario-based disk images considered in this paper.

Table 3
Public scenario-based disk images considered (source provided as clickable link).

Identifier	Year	Case Name	Origin
Bart23	2023	Bart the hacker	Instituto Politécnico de Beja
BelkaCTF1	2021	Insider Threat	Belkasoft
BelkaCTF5	2022	Party Girl Missing	Belkasoft
CCIKip	2019	Kip Case	California Cybersecurity Institute
CCITucker	2019	Tucker Case	California Cybersecurity Institute,
CellebriteCTF21	2021	Cellebrite CTF	Cellebrite
DefenitCTF20	2020	Find Tangential Cipher	Defenit CTF
DFRWSRodeo24	2024	DFRWS EU Rodeo Byte Busters	University of Zaragoza
Hadi1	n/a	Challenge 1- Web Server Case	Ali Hadi's Digital Forensics Challenge Images
Hadi2	n/a	Challenge 2 - User Policy Violation Case	Ali Hadi's Digital Forensics Challenge Images
Hadi3	n/a	Challenge 3 - Mystery Hacked System	Ali Hadi's Digital Forensics Challenge Images
InCTF20	2020	Lookout Foxy	InCTF Internationals 2020
LoneWolf18	2018	Lone Wolf Scenario	Digital Corpora
M57-08	2008	M57	Digital Corpora
M57-09Charlie	2009	M57	Digital Corpora
M57-09Jo	2009	M57	Digital Corpora
M57-09Pat	2009	M57	Digital Corpora
M57-09Terry	2009	M57	Digital Corpora
MagnetCTF19	2019	Magnet CTF	Magnet Forensics
MagnetCTF20	2020	Magnet CTF	Magnet Forensics
MagnetCTF22	2022	Magnet CTF	Magnet Forensics
MagnetCTF23	2023	Magnet Virtual Summit CTF	Magnet Forensics
NIST04	2004	Hacking Case	National Institute of Standards and Technology
OpenUni22	2022	Compromised Windows Server	The Open University
Owl19	2019	Owl	Digital Corpora

B. MDP plugin overview

Table 4 lists all currently supported plugins of MDP with further details.

Table 4
MDP Plugin Overview sorted by categories (Cat.) indicating Configuration, Activity, Longevity, Volume, Notables, with Cross-Platform property (CP) and Data Origin.

Plugin Name	Cat.	CP	Data Origin
disk_size	C	✓	Filesystem
total_sectors	C	✓	Filesystem
no_partitions	C	✓	Filesystem
(linux| mac| windows)_found	C	✓	Filesystem
win_build	C		Registry
win_registered_org_present	C		Registry
win_version_(id| str)	C		Registry
win_no_users	C		Registry
screen_(pixels| ratio)[e]	C		Registry
screen_resolution_(x| y)[e]	C		Registry
win_app_count_app_path_registry	C		Registry
win_app_count_uninstall_registry	C		Registry
(chrome| edge| firefox)_present	C		Registry
(chrome| edge| firefox)_default[b]	C		Registry
(earliest| latest)_fs_cr	L	✓	Filesystem
lifespan_fs_cr	L	✓	Filesystem
win_os_lifetime	L		Registry
win_install_time	L		Registry
win_last_shutdown_time	L		Registry
win_login_count_(max| total)	A		Registry
no_lnk_files_in_user_folders	A		Filesystem
no_recent_lnk_(max| total)	A		Filesystem
no_start_menu_lnk_(max| total)	A		Filesystem
no_prefetch_files	A		Filesystem
no_usb_mass_storage_attached_setupapi[d]	A		Setup API Log
firefox_searches_(max| total)[a]	A	✓	Firefox History
firefox_history_entries_(max| total)	A	✓	Firefox History
firefox_no_history_files	A	✓	Firefox History
chrome_searches_(max| total)[a]	A	✓	Chrome History

(continued on next page)

Table 4 (*continued*)

Plugin Name	Cat.	CP	Data Origin
chrome_history_entries_(max| total)	A	✓	Chrome History
chrome_no_history_files	A	✓	Chrome History
edge_searches_(max| total)[a,c]	A		Edge History
edge_history_entries_(max| total)[c]	A		Edge History
edge_no_history_files[c]	A		Edge History
browser_history_total	A	✓	Internet History
browser_searches_total	A	✓	Internet History
evtx_failed_logins_4625	A		Event Logs
evtx_success_logins_4624_2	A		Event Logs
evtx_unlocks_4624_7	A		Event Logs
evtx_clock_change_4616	A, N		Event Logs
no_files	V	✓	Filesystem
no_non_nsrl_files(_incl_zero)	V	✓	Filesystem
no_files_in_users_folder	V	✓	Filesystem
no_(audio| image| video)_files	V	✓	Filesystem
no_compressed_files	V	✓	Filesystem
no_(office| pdf)_files	V	✓	Filesystem
disk_usage_percent	V	✓	Filesystem
file_size_(mean| median| total)	V	✓	Filesystem
no_signature_mismatches	N	✓	Filesystem

[a] Separate plugins exist for searches conducted with different search engines (Google, Bing, and DuckDuckGo).
[b] Compatibility Restriction: Windows 7+.
[c] Compatibility Restriction: Edge version 79+.
[d] Compatibility Restriction: Windows Vista+.
[e] Compatibility Restriction: Windows 8+, some Windows 7+ systems.

References

Ballenthin, W., 2024a. Pure Python parser for Windows Event Log files. https://github.com/williballenthin/python-evtx. (Accessed 10 October 2024).

Ballenthin, W., 2024b. Pure Python parser for Windows Registry hives. https://github.com/williballenthin/python-registry. (Accessed 10 October 2024).

Boucher, J., Choo, K.K.R., Le-Khac, N.A., 2022. Web browser forensics – a case study with Chrome browser. In: A Practical Hands-On Approach to Database Forensics. Springer, pp. 251–291.

Breitinger, F., Jotterand, A., 2023. Sharing datasets for digital forensic: A novel taxonomy and legal concerns. Forensic Sci. Int.: Digit. Invest. 45, 301562.

Casey, E., Barnum, S., Griffith, R., Snyder, J., van Beek, H., Nelson, A., 2017. Advancing coordinated cyber-investigations and tool interoperability using a community developed specification language. Digit. Invest. 22, 14–45.

Chow, K.P., Law, F.Y., Kwan, M.Y., Lai, P.K., 2007. The rules of time on NTFS file system. In: Second International Workshop on Systematic Approaches to Digital Forensic Engineering (SADFE'07). IEEE, pp. 71–85.

Cohen, M., 2024. Python bindings for The Sleuth Kit. Accessed 10 October 2024. https://github.com/py4n6/pytsk.

Dreier, L.M., Vanini, C., Hargreaves, C.J., Breitinger, F., Freiling, F., 2024. Beyond timestamps: Integrating implicit timing information into digital forensic timelines. Forensic Sci. Int.: Digit. Invest. 49, 301755.

Du, X., Hargreaves, C., Sheppard, J., Scanlon, M., 2021. TraceGen: User activity emulation for digital forensic test image generation. Forensic Sci. Int.: Digit. Invest. 38, 301133.

Garfinkel, S., 2006. Forensic feature extraction and cross-drive analysis. Digit. Invest. 3, 71–81.

Garfinkel, S., 2024. Digital Corpora. Accessed 10 October 2024. https://digitalcorpora.org/.

Garfinkel, S., Farrell, P., Roussev, V., Dinolt, G., 2009. Bringing science to digital forensics with standardized forensic corpora. Digit. Invest. 6, S2–S11.

Göbel, T., Baier, H., Breitinger, F., 2023. Data for digital forensics: Why a discussion on 'how realistic is synthetic data' is dispensable. Digit. Threats: Res. Prac., 4 (3), 1–18.

Göbel, T., Maltan, S., Türr, J., Baier, H., Mann, F., 2022. ForTrace – a holistic forensic data set synthesis framework. Forensic Sci. Int.: Digit. Invest. 40, 301344.

Gruber, J., Humml, M., 2023. A formal treatment of expressiveness and relevance of digital evidence. Digit. Threats: Rese. Pract. 4 (3), 1–16.

Hargreaves, C., 2009. Establishing context when investigating a suspect's internet usage. In: Proceedings of 3rd International Conference on Cybercrime Forensics Education & Training (CFET 2009). Canterbury Christ Church University, 1st-2nd September 2009.

Hargreaves, C., 2017. Digital forensics education: A new source of forensic evidence. Forensic Science Education and Training: A Tool-kit for Lecturers and Practitioner Trainers 73–85.

Hargreaves, C., Breitinger, F., Dowthwaite, L., Webb, H., Scanlon, M., 2024a. DFPulse: The 2024 digital forensic practitioner survey. Forensic Sci. Int.: Digit. Invest. 51, 301844.

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

Menezes, A., van Oorschot, P.C., Vanstone, S.A., 1996. Handbook of applied cryptography. CRC Press.

Metz, J., 2024. Libewf. https://github.com/libyal/libewf. (Accessed 10 October 2024).

Moch, C., Freiling, F.C., 2009. The forensic image generator generator (Forensig^2). In: 2009 Fifth International Conference on IT Security Incident Management and IT Forensics. IEEE, pp. 78–93.

Moch, C., Freiling, F.C., 2012. Evaluating the forensic image generator generator. In: Digital Forensics and Cyber Crime: Third International ICST Conference, ICDF2C 2011, Dublin, Ireland, October 26-28, 2011, Revised Selected Papers 3. Springer, pp. 238–252.

NIST, 2024a. Computer Forensic Reference DataSet Portal (CFReDS). https://cfreds.nist.gov.

NIST, 2024b. Computer Forensics Tool Testing Program (CFTT). https://www.nist.gov/itl/ssd/software-quality-group/computer-forensics-tool-testing-program-cftt. (Accessed 12 October 2024).

NIST, 2024c. National Software Reference Library (NSRL). https://www.nist.gov/itl/ssd/software-quality-group/national-software-reference-library-nsrl. (Accessed 10 October 2024).

Patterson, J., Hargreaves, C., 2012. The potential for cross-drive analysis using automated digital forensic timelines. In: Proceedings from 6th Cybercrime Forensics Education and Training. Canterbury Christchurch University, Canterbury, UK.

Schmidt, L., Kortmann, S., Hupperich, T., 2023. Improving trace synthesis by utilizing computer vision for user action emulation. Forensic Sci. Int.: Digit. Invest. 45, 301557.

Schneider, J., Eichhorn, M., Freiling, F.C., 2022. Ambiguous file system partitions. Digit. Invest. 42, 301399.

StatCounter, 2024a. Desktop browser market share worldwide. https://gs.statcounter.com/browser-market-share/desktop/worldwide. (Accessed 10 October 2024).

StatCounter, 2024b. Desktop operating system market share worldwide. https://gs.statcounter.com/os-market-share/desktop/worldwide. (Accessed 10 October 2024).

Vanini, C., Hargreaves, C.J., van Beek, H., Breitinger, F., 2024. Was the clock correct? Exploring timestamp interpretation through time anchors for digital forensic event reconstruction. Forensic Sci. Int.: Digit. Invest. 49, 301759.

Voigt, L.L., Freiling, F., Hargreaves, C.J., 2024. Re-imagen: Generating coherent background activity in synthetic scenario-based forensic datasets using large language models. Forensic Sci. Int.: Digit. Invest. 50, 301805.

Wolf, D., Göbel, T., Baier, H., 2024. Hypervisor-based data synthesis: On its potential to tackle the curse of client-side agent remnants in forensic image generation. Forensic Sci. Int.: Digit. Invest. 48, 301690.