KAUNAS UNIVERSITY OF TECHNOLOGY

SANDEEPAK BHANDARI

RESEARCH AND IMPLEMENTATION OF TIMELINE ANALYSIS METHOD FOR DIGITAL FORENSICS EVIDENCE

Doctoral dissertation
Technological Sciences, Informatics Engineering (T 007)

Kaunas, 2022

This doctoral dissertation was prepared at Kaunas University of Technology, Faculty of Informatics Engineering, Department of Software Engineering during the period of 2017–2022.

The doctoral right has been granted to Kaunas University of Technology together with Vilnius Gediminas Technical University.

Scientific Supervisor:
Prof. Dr. Vacius JUSAS (Kaunas University of Technology, Technological Sciences, Informatics Engineering, T 007).

Edited by: English language editor Brigita Brasienė (Publishing House "Technologija"), Lithuanian language editor Aurelija Gražina Rukšaitė (Publishing House "Technologija").

Dissertation Defence Board of Informatics Engineering Science Field:
Prof. Dr. Rytis MASKELIŪNAS (Kaunas University of Technology, Technological Sciences, Informatics Engineering, T 007) – **chairperson;**
Prof. Dr. Nikolaj GORANIN (Vilnius Gediminas Technical University, Technological Sciences, Informatics Engineering, T 007);
Prof. Dr. Mirjana IVANOVIČ (University of Novi Sad, Serbia, Technological Sciences, Informatics Engineering, T 007);
Prof. Dr. Olga KURASOVA (Vilnius University, Technological Sciences, Informatics Engineering, T 007);
Prof. Dr. Simona RAMANAUSKAITĖ (Vilnius Gediminas Technical University, Technological Sciences, Informatics Engineering, T 007).

The official defence of the dissertation will be held at 10:00 a.m. on 9 June, 2022 at the public meeting of Dissertation Defence Board of Informatics Engineering Science Field in Dissertation Defence Hall at Kaunas University of Technology.

Address: K. Donelaičio St. 73-403, 44249 Kaunas, Lithuania.
Tel. no. (+370) 37 300 042; fax. (+370) 37 324 144; e-mail doktorantura@ktu.lt

Doctoral dissertation was sent on 9 May, 2022.
The doctoral dissertation is available on the internet http://ktu.edu and at the library of Kaunas University of Technology (K. Donelaičio St. 20, 44239 Kaunas, Lithuania) and http://biblioteka.vgtu.lt and at the library of Vilnius Gediminas Technical University (Saulėtekio al. 14, 10223 Vilnius, Lithuania)

KAUNO TECHNOLOGIJOS UNIVERSITETAS

SANDEEPAK BHANDARI

LAIKO ANALIZĖS METODO TYRIMAS IR ĮGYVENDINIMAS TEISINIAM SKAITMENINIŲ ĮRODYMŲ NAGRINĖJIMUI

Daktaro disertacijos
Technologijos mokslai, informatikos inžinerija (T 007)

Kaunas, 2022

Disertacija rengta 2017–2022 metais Kauno technologijos universiteto, Informatikos inžinerijos fakultetas, Programinės įrangos inžinerijos katedroje.

Doktorantūros teisė Kauno technologijos universitetui suteikta kartu su Vilniaus Gedimino technikos universitetu.

Mokslinis vadovas:
prof. dr. Vacius JUSAS (Kauno technologijos universitetas, Technologijos mokslai, Informatikos inžinerija, T 007).

Redagavo: anglų kalbos redaktorė Brigita Brasienė (leidykla „Technologija“), lietuvių kalbos redaktorė Aurelija Gražina Rukšaitė (leidykla „Technologija“).

Informatikos inžinerija mokslo krypties disertacijos gynimo taryba:
prof. dr. Rytis MASKELIŪNAS (Kauno technologijos universitetas, technologijos mokslai, informatikos inžinerija, T 007) – **pirmininkas;**
prof. dr. Nikolaj GORANIN (Vilniaus Gedimino technikos universitetas, technologijos mokslai, informatikos inžinerija, T 007);
prof. dr. Mirjana IVANOVIC (Novi Sado universitetas, Serbija, technologijos mokslai, informatikos inžinerija, T 007);
prof. dr. Olga KURASOVA (Vilniaus universitetas, technologijos mokslai, informatikos inžinerija, T 007);
prof. dr. Simona RAMANAUSKAITĖ (Vilniaus Gedimino technikos universitetas, technologijos mokslai, informatikos inžinerija, T 007).

Disertacija bus ginama viešame Informatikos inžinerija mokslo krypties disertacijos gynimo tarybos posėdyje 2022 m. birželio 9 d. 10 val. Kauno technologijos universiteto Disertacijų gynimo salėje.

Adresas: K. Donelaičio g. 73-403, Kaunas LT-44249, Lietuva.
Tel. (+370) 37 300 042; faks. (+370) 37 324 144; el. paštas doktorantura@ktu.lt

Disertacija išsiųsta 2022 m. Gegužės 9 d.
Su disertacija galima susipažinti interneto svetainėje http://ktu.edu, Kauno technologijos universiteto (K. Donelaičio g. 20, Kaunas, LT-44239, Lietuva) ir http://biblioteka.vgtu.lt, Vilniaus Gedimino technikos universiteto (Saulėtekio al. 14, 10223, Vilnius, Lietuva) bibliotekose.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

5WH – who, what, when, where, why and how
CFTL – cyber forensics time lab
DESO – digital evidence semantic ontology
DIALOG – digital investigation ontology
EPIC – explore, investigate and correlate
F-DOS – forensic-driven ontologies for smartphones
FOAF – the friend of a friend
GO – gene ontology
L2TCSV – log2timeline comma separated values
MACB – modified, accessed, change and create (birth)
NTFS – new technology file system
OSNs – online social networks
OWL – web ontology language
OWLViz – web ontology language visualization
RDF – resource description framework
SADFC – semantic analysis of digital forensics case
SKOS – simple knowledge organization system ontology
SPARQL – simple protocol and RDF (resource description framework) query language
SSDDF – small scale digital device forensics
TPFSM-A – temporal pattern of file system modification of the application
TPFSM-D – temporal pattern of file system modification of the hard disk
VOWL – visual notation for OWL ontologies

1. INTRODUCTION

Digital forensics is a process of identifying, collecting, preserving, analysing and presenting digital evidence that has been found on digital devices in the court. In order to attain the digital evidence during the investigation of digital crime, the reconstruction of the timeline is required. Moreover, the timeline assists in determining numerous activities that had been performed by a user on a particular system. The reconstruction of the timeline and interpreting the information from the timeline to collect digital evidence require an analysis of immense amount of events because of the explosive growth of the internet, interconnected devices, huge quantity of data, vast varieties of data, innovative technologies and many more. It has been found in literature studies that there are numerous approaches that have been developed for the reconstruction of a timeline to assist digital practitioners in understanding the timeline and interpreting the information and collecting digital evidence, but none of them were capable to address the challenges faced by the digital investigators, explore the evidence and understandability of the timeline in a competent way.

During the reconstruction of the timeline, the digital investigators encountered various new terminologies because of continuous innovations in technologies, the heterogeneity of data and many more issues. Moreover, the digital forensics tools generated an unstructured timeline from numerous sources of data. In such cases, the period that is required to find and interpret the cause of the potential digital incident can be affected by the complexity involved in understanding the meaning of newly encountered terminologies. In order to address these issues and assist digital investigators during the investigation of digital crime, two approaches have been developed that contributed in this field of research, i.e., first, the abstraction based approach for the analysis of timeline, and second, an ontology to define newly encountered terminologies during the analysis of the timeline.

1.1. Motivation

In this work, it was decided to develop a new abstraction based approach for the timeline analysis. The first reason was that the different existing approaches for reconstruction of timeline, developed by the researchers or authors, did not address the primary issues of digital investigation, such as automatic extraction of events and information from the timeline, heterogeneity, and huge volume of sources of data, clearly defined investigation model, capabilities of analysis and integrity of data. Another reason is that literature studies show that there are no available approaches for the reconstruction of timeline that are implemented on multiple operating systems based devices and where the results are evaluated. The third reason is that the concept of abstraction is not used for the analysis of the timeline in any existing approach.

In the developed approach, the analysis of the timeline is crucial for reducing the complexity of the timeline by splitting it into different and relevant levels of the timeline of events and artefacts. Thus, in the proposed approach, the timeline is split

into four relevant levels of timeline of events and artefacts, namely: Events: high level (new entries and web surfing), Events: low level (web surfing, actions of modifying), Artefact location: high level (include all application files) and Artefact location: low level. The main idea behind the breakdown of the timeline into four levels of abstraction is to present different kinds of information, and a different structure should be specified for each level along with distinctive levels of details of information to reduce the complexity of the timeline, omitting unwanted details, enforcing the correctness of timeline and presenting only information that will be helpful to recognise and understand particular actions executed by the users by analysing different sources and fields.

A novel ontology that is backed by an abstraction approach for timeline analysis has been developed and technically evaluated in this research work as well. The first reason was that there is no ontology found in literature studies that consists of basic terminologies of digital forensics domain and new terminologies, corresponding to different operating systems based devices. The second reason is that the existing ontologies are not technically verified and validated. The developed ontology consists of basic and new terminologies, corresponding to Windows, Android and iOS operating systems based devices. Moreover, the developed ontology is complete and easily expandable.

1.2. Object of the Research

The object of the research is the development of timeline analysis method and ontology that is based on the abstraction concept.

1.3. Aim and Objectives

The aim is to improve the digital forensics timeline by a novel method for the timeline analysis based on the abstraction concept.

Objectives of the thesis:

1. To analyse literature studies related to the basic terminologies, associated with the digital forensics domain, ontology and visualization along with distinct existing approaches that are available for timeline reconstruction, and ontologies in the digital forensics domain.
2. To develop a novel abstraction based approach for the analysis of timeline by defining four levels of abstraction of timeline and novel ontology by defining the basic and newly encountered terminologies.
3. To conduct numerous experiments and evaluate the research results of proposed approach and ontology.

1.4. Research Methodology

The research in this doctoral dissertation was performed by the utilization of methodology of Design Science research. Generally, the Design research methodology is used to develop a new artefact or enhance existing artefacts, such as algorithms, human/computer interfaces, design methodologies and many more [66].

In this research, there is an approach for timeline analysis, and there is an ontology for the digital forensics domain. The following Table 1 shows the methodology and various steps of this research.

Table 1. Research methodology

1. Analysing the domain in order to select and understand a relevant research problem.	From the analysis of digital forensics domain, it has been found that the reconstruction and analysis of the timeline is the fundamental problem for understanding various activities performed by the user and find the digital evidence.
2. Literature studies of existing approaches to understand their outcomes and find out a potential for the research.	The existing approaches are not mature to reconstruct and analyse the timeline to assist the digital investigator in understanding different activities and attaining digital evidence. Thus, a new approach has been developed.
3. Create a novel approach to solve the selected research problem.	A novel abstraction based approach has been developed. The novel approach consists of four distinct levels of abstraction of the timeline of events and artefacts. The proposed approach analyses timeline to identify different types of activities and their relevance performed by the user and reduces its complexity by splitting it into four different relevant levels and addressing various other issues.
4. Implement the approach and analysis of its outcomes.	The proposed approach is programmed in the object-oriented programming language Java and is implemented on different operating systems based devices, namely Windows, Android and iOS along with the analysis of outcomes.
5. Analysis of distinct existing ontologies of the digital forensics domain.	From the literature studies, it has been found that there are numerous ontologies that had been developed by different authors and researchers in the domain of digital forensics. All of them are goal-oriented, and they have been developed for specific goals and objectives in specific cases or scenarios. Moreover, the existing ontologies do not contain basic terms of digital forensic domain and new terminologies corresponding to various operating systems based devices. Since these ontologies cannot be applied to different cases or scenarios; thus, a novel ontology has been developed.

6. Develop a novel ontology.	The novel ontology is based on the timeline analysis of the proposed abstraction based approach. The novel ontology contains the definition of newly encountered terminologies during the analysis of the timeline of different operating systems based devices along with other basic concepts of the digital forensics domain. Moreover, the developed ontology is technically verified and validated.
7. Implementation of visualization.	The literature studies show that the digital forensics process can be enhanced by integrating information visualization techniques into the existing digital forensic investigation workflows that is known as Explore, Investigate and Correlate (EPIC) process. Thus, the visualization is implemented in the timeline analysis by an abstraction based approach to output an image to a user in a manner that facilitates the understanding of the underlying information.

1.5. Defended Statements

The statements defended by the dissertation are the following.

1. The novel abstraction based approach allows the analysis of the timeline that has been generated by digital forensics tools, namely Log2timeline and psort, by splitting the timeline into four relevant levels of the timeline of events and artefacts.
2. The novel ontology, backed by the proposed abstraction based approach for timeline analysis, allows digital practitioners or investigators to interpret basic and newly encountered terminologies of the digital forensics domain along with their significances.

1.6. Scientific Novelty

1. There has not been found any use of the abstraction concept for the analysis of timeline in any published research work, and the developed ontology consists of new terminologies, corresponding to Windows, Android and iOS operating systems based devices.
2. The novel abstraction-based approach reduces the complexity of the timeline by reconstructing it into four following levels: Events: high level, Events: low level, Artefact location: high level and Artefact location: low level. The approach is applicable to Windows, Android and iOS operating systems based devices.

3. The levels of the timeline are arranged in decreasing order of the abstraction, and each level adds additional information and details, omitting insignificant details and enforcing the correctness of the timeline.
4. The developed ontology consists of a description of newly encountered terminologies, corresponding to Windows, Android and iOS operating systems based devices. Moreover, the developed ontology is expandable, complete and concise.

1.7. Practical Applications

The developed solution facilitates an efficient way for the digital investigator or user to interpret different types of activities, performed by the user on digital devices. Moreover, the developed approach can be implemented on numerous operating systems based devices and allow to attain information from the timeline. A novel approach is implemented and validated on real data, i.e., Windows, Android and iOS operating systems based devices data. The outcomes show that the developed approach can be implemented for other operating systems based devices and used to collect information during the investigation of a digital crime efficiently by reducing the required time and manual labour.

The proposed ontology allows the digital practitioner or user to understand the primary terminologies of the digital forensics domain along with the newly encountered terminologies, corresponding to different operating systems and their importance. Moreover, the developed ontology is as well technically verified and validated, i.e., consistent, complete, concise and expandable. The new classes or concepts and definitions corresponding to them can be easily added without changing the already well-defined definitions in the developed ontology.

1.8. Results Approbation

Six articles and one abstract have been published on the topic of the dissertation. Two of the papers were printed in Web of Science indexed journals. Four papers and one abstract were published in scientific conferences in Lithuania and abroad.

1.9. Dissertation Structure

- The dissertation is divided into five chapters. The first chapter introduces the reader to the main research objectives and aim of the topic of digital forensics.
- The second chapter focuses on the digital forensics, ontology and visualization along with their fundamentals, available approaches for timeline reconstruction and existing ontologies in the digital forensics domain.
- The third chapter is dedicated to providing descriptive information about the proposed approach, based on the abstraction concepts and novel ontology for the digital forensics domain.

- The fourth chapter focuses on the implementation of the abstraction approach for the timeline analysis, novel ontology and visualization and their outcomes.
- The last section summarizes the whole work and gives main conclusions for this thesis and recommendations for the future work.

2. STATE OF THE ART ANALYSIS

2.1. Overview

In this section, an introduction and concepts or terminologies associated with digital forensics, ontology and visualization are discussed. The section consists of three major sections and numerous sub-sections.

- The "Digital forensics" section is devoted to presenting the concept of digital forensics, distinct major terminologies and scientific literature studies related to it. It consists of three major sub-sections, namely "main concepts of digital forensics", "digital forensic process" and "literature studies for the timeline reconstruction".
- "Ontology in digital forensics" section is devoted to presenting an introduction, concepts or terminologies associated with an ontology and its scientific literature studies in digital forensics along with the evaluation of ontology. This section consists of four major sub-sections, namely "main concepts of ontology", "constructing an ontology", "literature studies of approaches ontologies in digital forensics" and "ontology evaluation"
- "Visualization in digital forensics" section presents an overview of visualization and how visualization assists digital investigators to interpret digital evidence during the digital investigation process.

2.2. Digital Forensics

2.2.1. Main Concepts of Digital Forensics

During the last two decades, digital forensics has emerged because of the explosive expansion of the internet [92 & 98], the usage of electronic devices, rapid innovation in technology and growing size of storage devices and high rise in digital or computer crimes [108]. Distinct definitions are proposed by different authors, such as digital forensics (sometimes known as digital forensic science) is a branch of forensic science, encompassing the recovery and investigation of material found in digital devices, often in relation to computer crime [22 & 74]. The term "digital forensics" was originally used as a synonym for computer forensics [60] but has expanded to cover the investigation of all devices capable of storing digital data [69]. Prasad and Satish [108] defined digital forensics as the procedure of identifying, collecting, preserving, analysing and presenting digital evidence in a way that is legally accepted by the court.

Thus, digital forensics can be described as a series of discrete activities performed by different specialists of the digital forensics domain to understand the information and attain digital evidence from digital devices. Any device that can forward, receive and process digital information is a digital device, such as laptop, mobile phone, computer system and others. Although there are various and distinct reasons for performing digital investigation. Some of them are to attain and interpret digital evidence and present it to the court, identify a leak within an organisation and assess the possible damage that occurred during the breach. Using the data collected

[1]from electronic devices, digital forensic investigators can stop hackers and other cybercriminals from compromising organization's digital infrastructure. They can as well assist in recovering lost or stolen data, discover where a specific attack came from and trace it back to the source, and help create a detailed investigative report that can remedy any crime [68 &70].

Brady, Overill and Keppens [20] stated that digital investigators performed digital forensics by asking simple questions based on who, what, when, where, why and how? ('5WH') after the collection of information related to a digital attack by investigators that were investigating. The "5WH" process will assist the digital investigator to organize their information that they can extract knowledge and identify what action is needed next. It may not always be clear, however, exactly what the investigator is missing. By applying the 5WH formula to the material, the investigators can pinpoint specific gaps in a case, which may suggest potential lines of enquiry as shown in Figure 1.

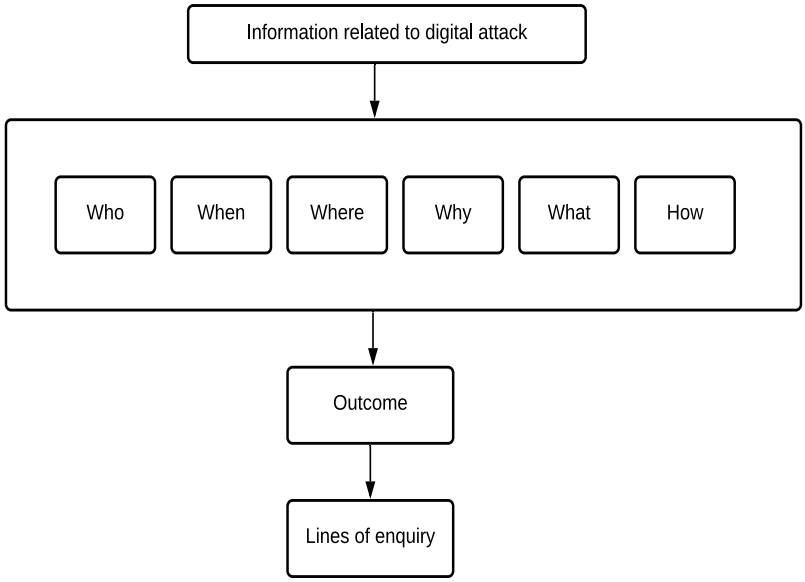


Figure 1. The reasoning for digital investigation (practice advice on core investigative doctrine [107])

[1] *Some passages have been quoted verbatim from the following source:*

An ontology based on the timeline of Log2timeline and Psort using abstraction approach in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

2.2.1.1. Digital Evidence

Digital evidence is any probative data stored or sent in digital form that is involved with, a legal dispute may use at preliminary. Before considering digital evidence, a court will decide whether the proof is important; a court will find out if the digital evidence is important, whether it is original, if it is hearsay and either a copy or original one is needed. The utilization of digital evidence has expanded in the previous few decades as courts have permitted to utilize messages, advanced photos, ATM exchange logs, word handling archives, text narratives, documents saved from bookkeeping programs, accounting pages, web program chronicles, information bases, the substance of PC memory, PC reinforcements, PC printouts, Global Positioning System tracks and computerized video or sound records [127, 119]. There are numerous origins of digital, yet it very well may be separated into three significant classes where digital evidence can be found to be specific Internet-based, independent PCs or gadgets and cell phones or mobile devices [111, 132 & 93].

2.2.1.2. Timeline[2]

There are distinctive definitions of the timeline provided by different authors in the digital forensics domain, such as the timeline can be defined as "a means of identifying or linking a sequence of events in a manner that is easy for people such as incident responders to visualize and understand" [25]. Harrell [63] defined timeline analysis as a great approach to identify various activities that have occurred on a particular system at a specific interval of time. Thus, constructing and analysing a timeline of numerous events or activities that occurred during an incident is one of the key tasks performed by the digital forensic practitioner [45, 23 & 16]. In the digital investigation, the analysis of the timeline is a fundamental component, as the timing of events or activities has nearly always been relevant. The primary source of timeline information is the file system metadata. File systems track different time stamps and have nuances that must be considered when performing forensic analysis [71, 16 & 106].

[2] *Some passages have been quoted verbatim from the following source:*

An abstraction based approach for reconstruction of timeline in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

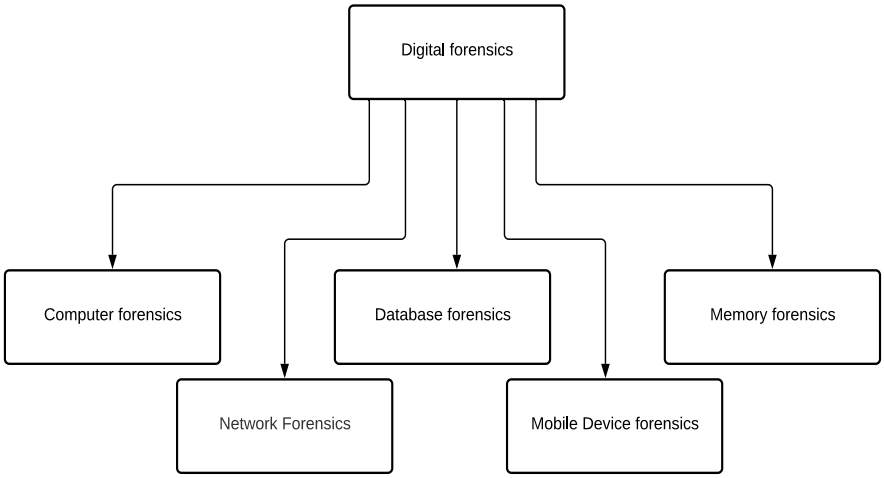


Figure 2. Types of digital forensics

2.2.1.3. Types of Digital Forensics

The digital forensic investigation is not limited to recovering information only from the PC, as laws are penetrated by the criminals, little computerized gadgets (for example, tablets, cell phones, flash drives) are broadly utilized at present [123].
A portion of these devices contains volatile memory, while some contain non-volatile memory. Adequate approaches are accessible to recover information from non-volatile memory, notwithstanding there is an absence of itemized technique or a structure for information recovery from volatile memory sources [88]. Contingent upon the sort of devices, media or artefacts, digital forensic investigation has expanded into different kinds, as demonstrated in Figure 2 [84].

2.2.1.3.1. Computer Forensics

Computer forensics is concerned with the identification, preservation, collection, analysis and reporting of evidences that are found on computers, laptops and storage media in support of investigations and legal proceedings. In computer forensics, digital investigators collect and extract various files on the systems, read the hard disk and find information from the computer to collect digital evidence [11& 45].

2.2.1.3.2. Network Forensics

The term network forensics has been utilized with expanding consistency for quite a while. Albeit, no official definition of network forensics exists, i.e., the term is regularly used to portray the assignment of breaking down data gathered on active networks from different interruption locations, inspecting and checking capacities for protection. The utilization of logically demonstrated procedures to gather, intertwine, recognize, look at, correspond, examine and record digital evidence from

various sources, effectively handling and communicating computerized hotspots to reveal the realities identified with the arranged purpose, or the estimated accomplishment of unapproved exercises intended to disturb, degenerate or potentially bargain framework segments just as giving data to aid reaction to or recuperation from these exercises [102 & 87].

2.2.1.3.3. Mobile Device Forensics

Mobile or Cell phone forensics is a part of digital forensics investigation, identifying with the recuperation of evidence or information from a cell phone under forensically solid conditions. The expression cell phone generally alludes to cell phones, notwithstanding it can likewise identify with any advanced gadget that has both interior memory and correspondence capacity, including PDA gadgets, GPS gadgets and tablet PCs [42, 134 & 126]. The utilization of cell phones/gadgets in wrongdoing was generally perceived for certain years, yet the scientific investigation of cell phones is a moderately new field, dating from the late and early 2000s. An expansion of phones (especially cell phones) and other advanced gadgets on the buyer's market caused an interest in the investigation of gadgets, which could not be met by the existing PC criminology procedures [32, 2 & 135].

2.2.1.3.4. Database Forensics

Databases assume a significant part in any association when capacity and registering segment materialize. Nowadays, all exercises are performed on the web and through which heaps of touchy and individual data get put away in the data set. Even though database security is not a novel method, there are still attacker attempts to alter the data set to remove such sort of data or attempt to erase it [12 & 4]. Database forensic is a field to address 5WH inquiries, for example, what, when, why, where, how information base altering has occurred and by whom. It is a subfield of digital forensics investigation, which centres around the definite investigation of a data set, including its substance, log documents, metadata, and information records, relying upon the sort of data set utilized [38 & 29].

2.2.1.3.5. Memory Forensics

Memory legal sciences is an essential type of digital examination that permits a digital practitioner to distinguish unapproved and irregular action on an objective PC or worker. This is typically accomplished by running special programmes that catch the present status of the framework's memory as a preview document, otherwise called a memory dump. This document would then be able to be taken offsite and looked at by the examiner. This is helpful due to the manner by which cycles, documents and projects are run in memory, and once a preview has been caught, numerous significant realities can be found out by the specialist, for example, Processes running, Executable records that are running, Open ports, IP addresses and other systems administration data, Users that are signed into the framework, and from where Files that are open and by whom [72 & 89].

2.2.1.4. Challenges in Digital Forensics

In the current situation, digital violations have immersed the digital world, making a few difficulties for network protection specialists. Since the absence of mindfulness

among the end clients makes a way for attackers to misuse them or their associations, digital forensics has acquired tremendous importance in the examination cycle of an incident identified with cybercrime [118]. This has likewise made many issues and difficulties for the specialists and analysts, some identified with innovation or progression, some identified with guidelines and rules and some identified with the essential usefulness of the examination [71 & 91]. The fundamental issues and difficulties of digital forensics can be characterized primarily in three significant parts as demonstrated in Figure 3.

2.2.1.4.1. Source Related Issues and Challenges

These sorts of difficulties and issues come in digital forensics due to the functional issues. These issues are identified with the fundamental climate or strategy that is taken by the specialists and analysts to examine the occurrence [136 & 90]. A portion of significant issues are scalability and collection and visualization of digital evidence [12, 28 & 110].

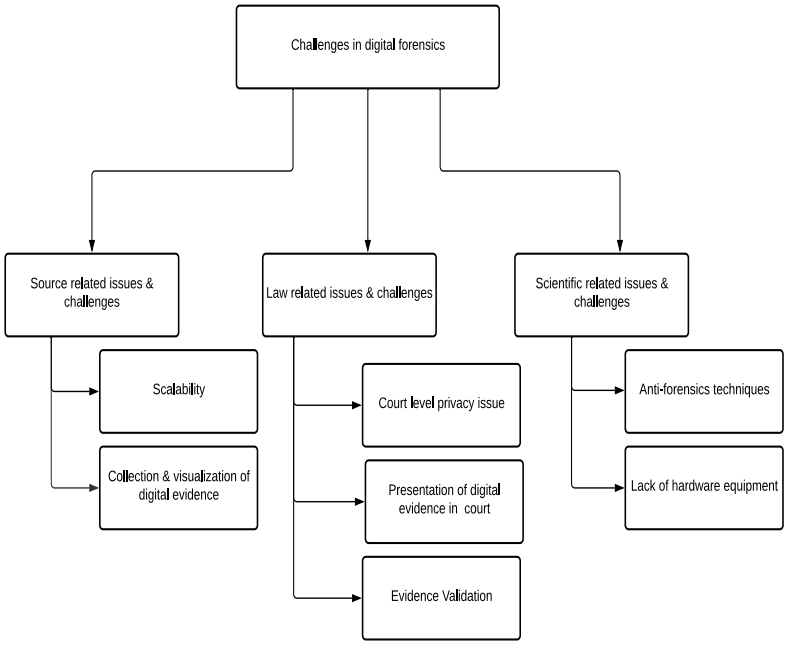


Figure 3. Challenges in digital forensics (Pandey et al. [103])

2.2.1.4.2. Law Related Issues and Challenges

In this kind of issue, various nations have various laws, and a few nations do not have a law or set up principles for digital and scientific assessments [49 & 26].

Thus, there are a few ambiguities and issues identified with digital forensics investigation and online protection laws. For example, if an investigator tracks down that a particular digital crime has been performed by a system that is situated in an outside country and that nation does not have any digital law, at that point, the investigator cannot do anything, and this makes it a huge test for the specialists and analysts. There are numerous other law-related issues. A portion of significant issues are court level privacy issues, presentation of digital evidence in court and evidence validation [128, 112 & 9].

2.2.1.4.3. Scientific Issues and Challenges

These are basic issues in the present period, because the utilization of innovation is available in both decent and awful manners. Like digital investigators and other specialists utilize the PCs and advances decently to analyse the proof and crime location, a few people utilize the innovation and PC in a useless way to do some illicit, unapproved movement and to be unknown. This sort of utilization of innovation and PCs deductively make an issue, and this is both the most risky and the best issue of the present period. There are basically two sorts of difficulties, namely, anti-forensic techniques and lack of hardware equipment [54].

2.2.2. Digital Forensic Process

The Internet, PC organizations and robotized information frameworks present a huge new chance for carrying out a crime. PCs and other electronic gadgets are being utilized progressively to carry out, empower or support wrongdoings executed against people, associations or property [26]. Regardless of whether the wrongdoing includes assaults against PC frameworks, the data they contain or more customary violations like homicide, illegal tax avoidance, dealing or extortion, electronic proof progressively is included. It is nothing unexpected that law implementation and criminal equity authorities are being overpowered by the volume of examinations and indictments that include an electronic proof. In order to help the State and nearby law authorization offices and prosecutorial workplaces with the developing volume of electronic wrongdoing, a progression of reference guides in regards to practices, techniques and dynamic cycles for examining electronic wrongdoing is being set up by specialized working gatherings of professionals and topic specialists who have learned about electronic wrongdoing.

One of them is the digital forensics process, which defines the investigation process, starts from the crime scene first responder to the laboratory, to the courtroom. The nature of electronic evidence is such that it poses special challenges for its admissibility in court. In order to meet these challenges, the following proper forensic procedures are called the digital forensics process. These procedures include, but are not limited to four phases: collection, examination, analysis and reporting [90]. The digital forensic process is a recognized scientific and forensic process used in digital forensics investigations [28 & 120]. The digital forensics process is multi-staged, beginning from the identification of digital devices from the scene, as potential evidence to the stage, where it is presented as evidence of an

expert witness in a court of law [114 & 138]. Generally, a digital forensics process contains five major stages, namely identification, preservation, examination, analysis and documentation and report to present the digital evidence in the form of the law of court, as shown in Figure 4.

2.2.2.1. Identification

The digital forensics process begins with the first stage that is called the identification of digital evidence. In this phase, all potential sources of relevant evidence/information (devices) at the crime scene are identified and collected. It includes the distinguishing proof of computerized gadgets fit for putting away advanced information related to the examination. A few examples that can give electronic proof such as the hard disk of PC frameworks, irregular (random) access memory cards, USB and other external sources of secondary memory such as cell phones, USB and many more [24].

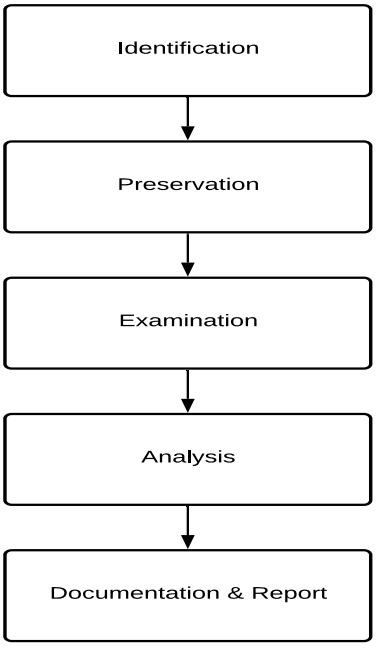


Figure 4. Digital forensics process

2.2.2.2. Preservation

After the identification of potential sources of digital evidence at the first stage of digital forensics, the sources need to be preserved to collect relevant information or evidence and maintain the integrity of evidence. Moreover, distinct investigation methods are employed by digital practitioners that do not alter the digital evidence or data. The most common is a chain of custody, Drive Imaging and Hash Values.

2.2.2.3. Examination

The examination of digital evidence from the identified possible sources seized at the scene of a crime is the third phase of the digital forensics process. In digital forensics, the examination of digital evidence is about the recovery and extraction of data from various available digital sources. There are two distinct sorts of extraction, i.e., physical and legitimate. The physical extraction stage recognizes and recuperates information across the whole actual drive, regardless of the file system. The legitimate extraction stage distinguishes and recuperates documents and information dependent on the introduced working system(s), record system(s) or potentially application(s) [13 & 97].

2.2.2.4. Analysis

In the analysis phase, the digital investigator performs the interpretation of extracted digital data and tries to determine their relationship and their significance with digital crime. Moreover, the investigators as well try to find digital evidence by analysing the interpreted data that either supports or contradicts a hypothesis. Reith, Carr and Gunsch [113] defined the evidence analysis phase as "an in-depth systematic search of evidence related to the suspected crime".

2.2.2.5. Documentation and Report

The last phase of the digital forensics process is documentation and report. In this stage, the digital practitioner should report totally and precisely each progression in their examination from the beginning as far as possible. The point is to permit others following the means laid out in the documentation to imitate the examination and arrive at similar resolutions. At the point when an examination is finished, the data is frequently revealed in a structure that is appropriate for non-specialized people.

2.2.3 Literature Studies of Approaches for the Timeline Reconstruction

Esposito and Peterson [47] highlight the importance of timeline in digital forensics investigation, and the stated timeline is a valuable approach to identify the distinct events or activities that are performed by the user on a particular digital device during a specific period. Thus, the construction and analysis of the timeline are one of the primary tasks that are carried out by the digital investigator. Sitompul, Handoko and Rahmat [123] state that it is challenging to have a clear view of events that occurred during a time period in the digital investigation. Event reconstruction, which allows a digital investigator to understand the timeline of a crime, is one of the paramount steps of the digital investigation process [37]. This complex task requires exploration of a large number of events due to the technology innovation frequently, a heterogeneous, huge quantity of data and a manually performed event reconstruction process, which is inefficient and expensive [76, 35] [16]. Inglot and Liu [70] specified that there are basically two approaches for the analysis of the timeline. Firstly, some applications have been specifically created for the analysis of the timeline, and they focus on visually presenting the timeline, such as Cyber

Forensics Time Lab (CFTL), Zeitline, Encase, Sleuth kit, Forensic toolkit, and many others [110] [16]. Secondly, there is a combination of command-line tools and spreadsheet applications that are labour intensive, such as Log2timeline and excel together.

Guðjónsson [59] designed a tool to extract timestamps from various files found on a typical computer system and aggregate them. The tool Log2timeline is a part of a Python-based backend engine plaso. The purpose of plaso was to have the timestamps in a single place for computer forensic analysis. Such a timeline sometimes is called a Super Timeline. Log2timeline as well addresses numerous problems of traditional timeline analysis approach, such as easy to manipulate, prone to change, not frequently updated and labour intensive by performing automatic extraction of the timeline. However, there are still various issues in the timeline generated by Log2timeline, it includes a huge number of events, differences in cases that required manual inspection of the timeline. These issues can be handled by finding a way to reduce the dataset (events) in an intuitive method.

Sitompul, Handoko and Rahmat [123] highlight the importance of file recovery in computer forensic investigation. Some factors need to be considered during the recovery of a deleted file, such as a deleted file may be potentially modified from its original status by another file partly or completely. For this, the authors mention that there is an approach to the recovery of deleted files, i.e., Boyer-Moore algorithm, but it has high time complexity in terms of string searching. Thus, they propose a better string matching approach for the recovery of a deleted file that is known as Aho-Corasick parsing technique. This approach works in four steps, i.e., disk imaging, accessing MFT (master file table), file type identification and corruption check and file reconstruction (undelete, verification and analysis). Further, they implement this approach on 3.54 GB data that consists of 56 files of various types, such as .docx, .pdf, .jpg, .png, and .exe files. The proposed approach is able to successfully recover 55 files (98.21%) from 56 files in 229.418 seconds with an average data processing speed of 15.77 MB/s. The major drawback of this approach is that it entirely depends upon master file table condition, i.e., if the MFT is damaged, it will affect the recovery output.

Bang et al. [10] have discussed how the creation time, last written time and last accessed time of a file or folder are important factors that can indicate events that have affected the computer system. They analysed changes in the time information of files and folders for different operations of the FAT and NTFS file systems and attempted to reconstruct the user's actions. For this, they implement the approach based on the experiences collected from digital forensic investigations: frequently occurred events are reasoned with the essences of MAC times. These events or activities have a high prospect in event reconstruction; some phenomena are observed and studied for creating heuristic rules (7) in the MAC times analysis. These rules are expected to assist computer forensic examiners in investigating the digital events that have occurred. Further, they demonstrate the use of time information for digital evidence analysis by presenting a case study of data manipulation in LC/MSD(Liquid Chromatography-MASS Selective Detector). The

authors have found from the results that the proposed approach is capable to retrieve the time information related to different operations, performed by the user on the windows operating system based computer system.

Chabot et al. [31] have identified two major challenges (heterogeneity and volume of data) with event reconstruction. In order to solve these two challenges, they present an approach supported by the theoretical concepts that can assist investigators through the whole process, including the construction and interpretation of the events, describing the case that is known as SADFC (Semantic Analysis of Digital Forensic Cases). The proposed approach is based on a model, which integrates knowledge of experts from the digital forensic fields and software development to allow a semantically rich representation of events related to incidents [16]. The SADFC approach consists of three different modules, namely advanced timeline analysis model, investigation process model and ontology-centred architecture. The major limitation is that this approach is still underdeveloped as only the first module (advanced timeline analysis model) of the approach is developed, and the remaining two modules need to be developed. Thus, the performance can be analysed after the implementation of the approach for the automatic reconstruction and analysis of events.

Hargreaves and Patterson [62] focus on the issues of the super timeline generated by Log2timeline digital forensic tools, such as a number of events that makes the analysis hard and limit the way in which the data can be visualised. Thus, they developed a software prototype known as Python Digital Forensic Timeline (PyDFT) by using Python 3 language to combine 'low-level' events (i.e., data extracted from the file systems and compound files) into 'high-level', which are human understandable events automatically. The developed software works in two main phases, i.e., the generation of low-level event and the reconstruction of high-level event. The low-level events are extracted from the inside files, using an 'extractor manager', and are converted into a standard format for a low-level event, then added to a timeline. This timeline is stored as a SQLite database, which can be used for further queries. The high levels are reconstructed from the low level events by searching for patterns of events in the low-level timeline based on the pre-determined rules. These patterns are based on a plugin framework where each plugin is a script that detects and reconstructs a particular type of high-level event. Further, they show two case studies of generating two high-level events, i.e., detection of Google searches and connection of a USB device to show the implementation of the developed software prototype. The shortcoming of the approach was that the patterns, which were oriented to particular events, have to be written in advance [16].

Brady, Overill and Keppens [19] proposed the use of ontology, the Digital Evidence Semantic Ontology (DESO), which allows an examiner to quickly discover what artefacts may be available on a device before time-consuming processes are commenced. The DESO is built on the ideas of Gene Ontology (GO). The general principle behind DESO is twofold: (1) examiners use some form of classification or tagging system that allowed examiners to readily assess what

artefacts were available; (2) once artefacts have been extracted from various sources, DESO provides the means to compare them. The main idea of DESO is to enable the comparison of the artefacts extracted from different sources. Brady, Overill and Keppens [20] continued the development of the ontology DESO. DESO's primary purpose is to act as a repository and a classifier of digital evidence artefacts to allow the correlation of extracted data from heterogeneous sources. The investigative objectives are set and fulfilled by asking simple questions based on who, what, when, where, why and how? ('5WH'). Only the "What" subclass has been detailed. The classes "why" and "how" were not discussed at all. An implemented body of DESO was not revealed. It remained behind the scenes. Only the ideas were presented [16].

Debinski, Breitinger and Mohan [43] state that event reconstruction is a fundamental step for investigators to understand a case where a prominent tool is Log2timeline to generate timelines. While these timelines provide great evidence and assist to understand a case, they are complex and require tools as well as training scenarios. Moreover, they as well state some of the major limitations of Log2timeline, such as the fact that there is no easy-to-use tool that beginners/investigators can use to analyse a generated timeline and no free training material that allows practitioners to learn and improve their familiarity with the Log2timeline as well as visualization tools. In order to support the investigators, the authors developed Timeline2GUI a standalone tool written in Python that supports the analysis of the CSV timeline (output from Log2timeline). The main goal of developing Timeline2GUI is to compose the parsing (reading) of the log files straightforward for the end-user. The graphical user interface remains simple and is based on a commonly used excel sheet. There are two views in the Timeline2GUI for the effective analysis of the timeline, namely reduced view and detailed view. The reduced view illustrates the highlighted events, and the detailed view shows the complete timeline. Besides, they developed three training cases that are freely available and can be used to improve the investigator's timeline analysis skills by either using Timeline2GUI, the Excel sheet or any other tool. From the results of the training cases, it has been found that Timeline2GUI is beneficial for the digital investigator to understand the timeline and attain digital evidence by using its different features, such as filtering of the events based on the date and time of the event, searching capabilities of a particular type of files and many more. However, the performance of Timeline2GUI is its major limitation, but it can be enhanced by speeding up the process, such as by removing irrelevant fields or combining fields in future work [16]. Soltani, Seno and Yazdi [124] proposed an event reconstruction framework that determines whether an application has been run on a compromised system. The proposed framework has constructed the signature or the TPFSM-A (temporal pattern of file system modification of the application) and the TPFSM-D (temporal pattern of file system modification of the hard disk). Moreover, the framework has presented a distance metric that is used to calculate the distance between the signature of the application and TPFSM-D of the hard disk. Finally, the decision engine of the framework has used the calculated distance to decide whether

the application has been run on the compromised system. In order to demonstrate the capability of this approach, the implementation is performed into two-phase, namely the training phase and the detection phase, to find out whether the applications, namely Microsoft Word, Adobe Reader, Firefox and Windows Media Player, are executed on a particular computer system. The results illustrate the effectiveness of the proposed framework in reconstructing events. The precision and accuracy of the proposed framework reached 94%. The major limitation of this approach is that it focuses on and is suitable for Windows operating system timeline reconstruction.

Forensic Toolkit (FTK) is a computer forensics tool developed by the Access Data Group [51]. FTK examines the hard drive and retrieves information, such as deleted emails, text strings, to use them as password dictionary to unlock the encryption. FTK is as well associated with a standalone disk imaging tool known as FTK imager. This tool captures the image of a hard disk in a file that will be reconstructed later. FTK use MD5 and SHA1 hash value to verify if the integrity of the data imaged is consistent with the created forensic image. FTK imager creates a bit-for-bit duplicate image of the media to avoid the manipulation of the original evidence accidentally or intentionally. The forensic image of the evidence is similar to the original in every perspective, including the file stack and unallocated drive space. This will allow the digital investigator to keep the original media away and safe from any kind of harm, while the investigation proceeds by using the image. After the creation of the image, the digital investigator can use the Forensic Toolkit (FTK) to perform a complete forensic investigation and generate a report of digital evidence. FTK has various features, such as a simple user interface and advanced searching capabilities, supports EFS decryption, generation of the case log file and bookmarking. However, it has some drawbacks as well, such as it does not provide scripting features, multi-tasking capabilities, a progress bar to estimate the time remaining and a timeline view.

Encase is a digital forensics tool developed by Guidance software [46]. Encase is developed for forensics, digital security, security investigation and e-discovery usage. It is commonly used to recover digital evidence from seized hard drives or other digital devices. Encase assists the digital investigator or specialist to direct from beginning to end the digital investigation of client record to collect digital evidence that can be used in the court of law. The encase digital forensic tool follows the basic digital forensics investigation process that begins with the acquisition of digital devices, analysis of data and ends with the presents of digital evidence in the form of a report. The numerous feature of Encase includes a collection of data from a wide variety of sources, generation of large scale of reports on finding along with maintaining the integrity of digital evidence, usage of keyword, metadata and hash values on target sources of evidence, disk imaging, data carving and password recovery. However, it has some drawbacks as well, such as it is a very expensive tool, has high processing time in case of large compound files and mailboxes and is not compatible with other forensic based tools.

2.2.3.1. Research on Digital Forensics in Lithuania

The authors stated that the acquisition of digital remnants and their use to find crime footprints in the digital user places (device, profile, home directory, etc.) is challenging [54 & 55]. They as well highlight the benefits of the use of digital profiles in the investigation, such as linking together different crimes, narrowing down the lists of suspects, aiding the process of investigation, reconstructing the crime and adding confidence to the existing evidence, while a profile cannot be used as direct evidence. This simplifies the investigation process and makes use of profiling to have a successful result. Thus, they propose a model for digital evidence investigation based on the habits attribution that is known as the habits identification domain (HiD) model. The main idea was to identify habits, attribute them and then create a profile of the attributed habits. The created profile, as a set of habits and attributes, may be used in digital evidence investigation to reduce the numbers of evidence search sequences from a set of digital user places. It analyses data and metadata memorized into a digital device by applying specific techniques taken from intelligence and traditional profiling to obtain information that helps to create a digital profile with suspect user habits attributes and then consider it during the evidence investigation. In order to demonstrate the capability of HiD model, the authors implement the HiD by using four different functions on a case study, having two user profiles: the first profile coincides with all files on the user's a hard disk drive and the other profile coincides with all files on the user's hard disk snapshot. The major drawback of this paper is that there is no comparative analysis with other models based on the habit attribution or any other similar criteria that are discussed by the author to show the performance of HiD model.

In this paper, the author creates an ontology-based transformation model and a framework to develop an ontology-based transformation system (OBTS) in the digital forensics domain [56]. The authors as well define the architecture of the ontology-based transformation system and its components for assisting computer forensics experts in the appropriate selection of tools for digital evidence investigation. The authors analyse two domains, namely cyber forensics ontology (CFO) and Computer Forensics Tool Catalog (CFTC), developed by the National Institute of Standards and Technology (NIST). They have found that these two domains have created common definitions in the digital forensics domain. While both belong to the same digital forensics domain, they are very different, and only a small number of artefacts, when expressed through ontologies, intersect. Typically, computer forensics experts operate in terms of the CFO, but the NIST taxonomy of forensic tools for digital evidence investigation is given in CFTC terms. In order to handle this issue and facilitate the computer forensics experts in selecting an appropriate tool for digital evidence investigation, the authors propose a computer forensics tool catalogue ontology (CFTCO) created from the NIST CFTC and an ontology-based transformation model (TM) for the digital forensics domain. An ontology-based TM consists of two stages. The first stage relies on the XML view creation for the selected ontologies (CFO and CFTCO). In the second stage, the

Table 2. Comparative studies of available timeline approaches

Timeline reconstruction approaches	Auto extraction	Heterogeneity and volume	Analysis	Theory	Integrity
ECF [35]	✓	✓	×	×	×
Guðjónsson [59]	✓	✓	×	×	×
Sitompul, Handoko and Rahmat [123]	✓	×	×	✓	×
Bang et al. [10]	✓	×	×	×	×
Hargreaves and Patterson [62]	✓	✓	×	×	×
Debinski, Breitinger and Mohan [43]	✓	✓	×	×	×
Soltani, Seno and Yazdi [124]	✓	×	×	✓	×
Forensic Toolkit [51]	✓	✓	×	×	✓
CyberForensic TimeLab [98]	✓	✓	×	×	×
Encase [46]	✓	✓	×	×	✓
Zeitline [31]	×	✓	×	×	✓
Esposito and Peterson [47]	✓	×	×	✓	×

transformation process uses the XML view of the ontologies created in the first stage and maps their representations from one form to the other. The transformation process applies a set of transformation rules that will create a list of appropriate tools. The authors present a case study of transforming CFO to NIST tool list by using a set of transformation rules in the OBTS and assist computer forensics experts in selecting an appropriate tool for further digital evidence investigation. The main drawback of this paper is that there is no performance analysis of an ontology-based transformation system (OBTS) that is discussed by the author, and the output of CFTCO is bound to CFO and CFTC.

The literature studies show that the reconstruction and analysis of the timeline have numerous issues, such as the volume of data, the heterogeneity of data, the complexity of digital investigation process and many more. Moreover, in order to address all these issues and assist the digital forensic investigation process, numerous approaches and tools have been developed from time to time. Table 2 shows the comparison of various existing approaches ((✓) shows the strengths of approaches, and (✗) shows the weakness of the approaches) with relation to some primary issues in digital forensics investigation, namely heterogeneity and huge volume of sources of data, automatic extraction of events and knowledge from the timeline; a clearly defined investigation model allows to describe a process used to get the required results, analysis capabilities and maintain the integrity of data.

The comparative studies show that most of the approaches can address issues, such as automatic extraction of events and generating the timeline, processing huge volumes and different sources of data and organizing the information in a systematic way. The other key issues, i.e., a clear description of used investigation model in the reconstruction process, are required to ensure the reproducibility of the investigation process and the credibility of the results, which is satisfied by a few of the existing approaches. The integrity of the collection of data from different sources from crime scenes and the integrity of extracted digital evidence from the collected data should be maintained, which is one of the key issues in the digital forensic investigation and is not fulfilled by most of the available approaches. The analysis of the timeline, i.e., to assist the digital investigator or user in understanding the timeline, composing timeline easy to read, recognizing the correlation between events, producing a conclusion in the form of digital evidence from the knowledge contained in the timeline, is an issue that not addressed and fulfilled by available approaches.

The majority of the existing approaches can address only two issues, i.e., auto extraction of events and timeline and processing huge volume and different sources of data. Moreover, some of the approaches are as well able to fulfil the other two key issues, i.e., the integrity of data and a clear description of the used investigation model. Although the analysis of the timeline issue is not addressed by the available approaches to assist digital practitioners in the investigation process to understand the timeline effectively [83, 39 & 121], some of the primary reasons, which are responsible for the ineffective analysis of timelines, include the rapid innovation of technology, accelerated growth of the internet and integrity of the timeline [16]. In

order to address this issue, a novel approach backed by an abstraction approach is proposed.

2.3. Ontology In Digital Forensics

2.3.1. Main Concepts of Ontology

The word ontology comes from the Greek ontos (being) + logos (word). The Merriam Webster online dictionary defines the term ontology [99] as a part of metaphysics, worried about the nature and relations of being, a specific hypothesis about the idea of being or the sorts of existents. The term ontology was presented as a way of thinking, in the nineteenth century, by the German savant Rudolf Gockel, in his *Lexicon Philosophicum*, to recognize the investigation of “being” from the investigation of different sorts of creatures in the common sciences. As a philosophical control, ontology building is worried about giving classification frameworks that record to a specific vision of the world. The previously known classification framework was proposed by Aristotle. In his framework, a classification is utilized to group whatever it can be said about anything. In the third century BC, Porphyry, a Greek scholar, remarked on Aristotle's construction and coordinated the proposed classes in a tree graph. This construction, known as the Tree of Porphyry, is demonstrated in Figure 5. According to Elhadad, Badran and Salama [44], ontology is "the investigation of classes of things that exist or may exist in some area. The result of such examination is called an ontology is an inventory of sorts of things that are expected to exist in an area of interest D from the viewpoint of an individual who utilizes a language L to discuss D." Ontology is based on the facts and its nature, being independent of one’s background, showing understanding, perspective and knowledge of the world [61 & 122]. First, the artificial intelligence researchers used the ontology approach from philosophy, as shown in Figure 6 [115]. Since then, the concept of ontology has been used by the scientists from information and computer fields. The explanation of ontology most regularly cited in semantic web writing is the following: "An ontology is a formal, explicit specification of a common conceptualization" [60]. In this case, the conceptualization represents a theoretical model, explicit implies that the components should be obviously characterized, and formal shows that the detail ought to be machine-processable. Going further, in Gruber's view, an ontology is the portrayal of information on a domain, where a bunch of objects and their connections is depicted by a vocabulary. By and large, a gathering of scientists who need to share data in a specific area builds up an ontology. There are various explanations behind building up an ontology, including basic understanding, reusing, sharing, breaking down the domain information, detachment of area information from operational information and making area suppositions expressly [55].

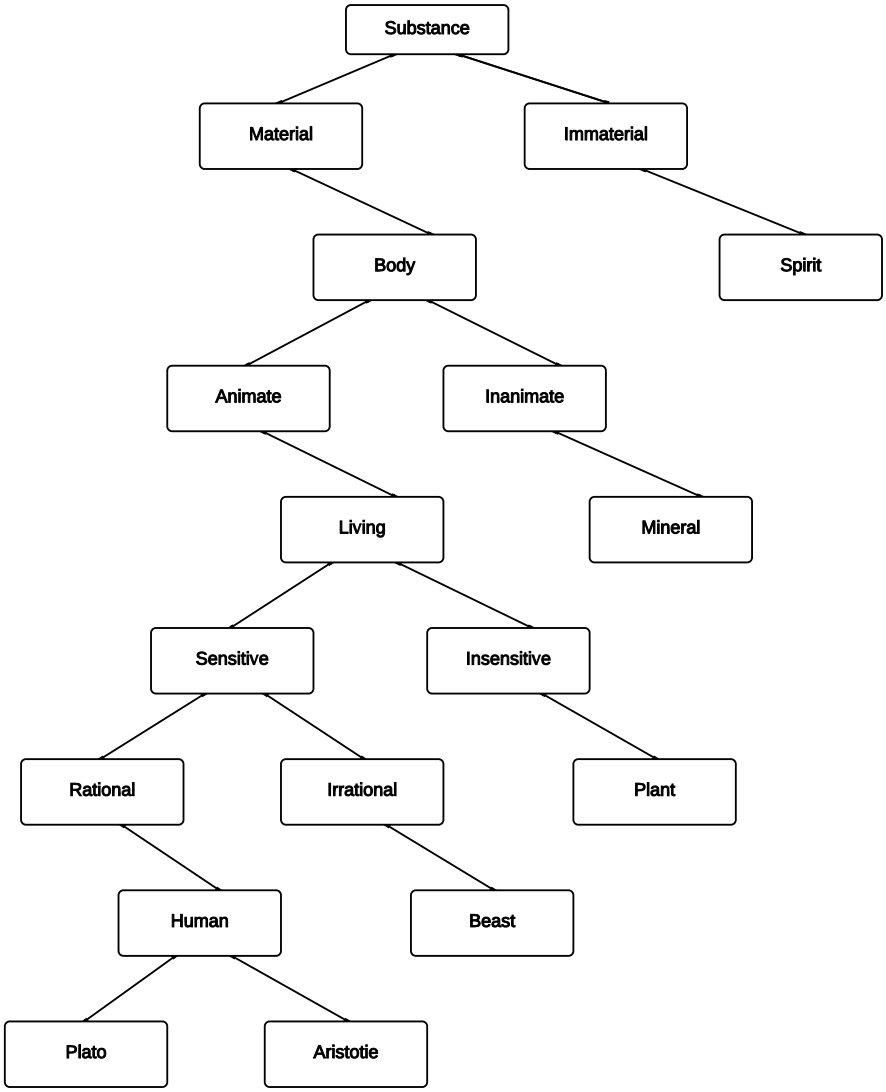


Figure 5. Tree of porphyry (Ontology in Computer Science [99])

A particular clarification of ontology is accessible in PC and data science writing. In any case, all analysts concur on the significance of ontology in the portrayal, sharing and reusing of existing area information [59]. The most well-known meaning of ontology among specialists is an assemblage of officially addressed information, depending on the conceptualization. A conceptualization is theoretical, worked visible of the world that people wish to address for some reason. Each information base, information-based framework or information level specialist is focused on some conceptualization, expressly or certainly. An ontology is an

unequivocal particular of the conceptualization [56]. In the digital forensic area, it is not feasible to build up an ontology that would be adequately huge to contain every idea that occur, and which are important to individuals who lead the digital forensics investigation [60, 96]. An ontological portrayal of the gathered information can take care of the issue of a variety of information in digital forensics [7, 5].

Figure 6. The shift of ontology to computer science field (Russell, Norvig [115])

2.3.1.1. Components of an Ontology

A typical ontology of a domain consists of different components, namely classes, instances, properties and relations, as shown in Figure 7.

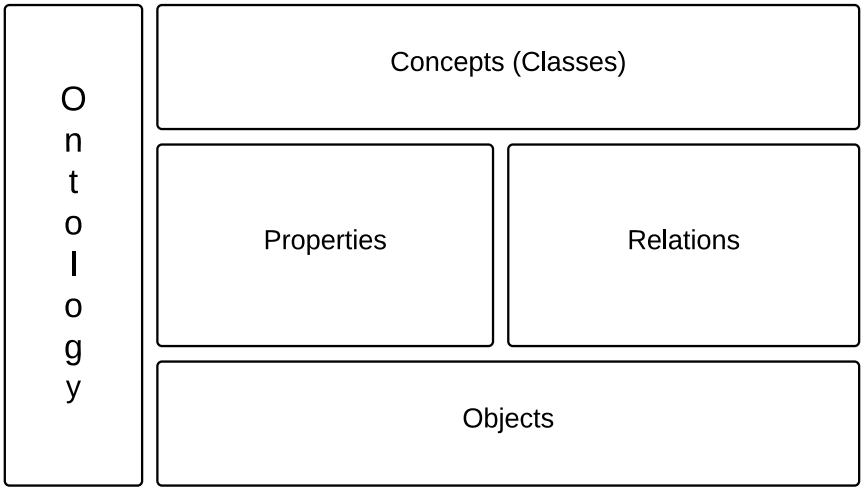


Figure 7. Components of an ontology [15]

2.5.1.1.1. Classes

Concepts, likewise called classes, types or universals are a central part of most ontologies. A concept addresses a gathering of various people that share normal attributes, which might be quite explicit. For instance, (most) people share certain attributes, like related DNA, a bunch of explicit body parts, the capacity to communicate in an unpredictable language. In such manner, all vertebrates share these qualities, with the exception of the capacity to talk. One concept might be a subconcept of (otherwise called a subclass, or sort of) another concept; this implies that assuming the idea C' is a subconcept of C, any person of type C' will likewise be a person of type C. It is conceivable inside an ontology to unequivocally express that C' is a subconcept of C; in certain dialects, including OWL, it is likewise conceivable to deduce this. The concepts may likewise impart connections to one another; these portray the path to people of one concept to identify with the people of another.

2.3.1.1.2. Instances or Objects

Individuals (objects) are the essential, "ground level" segments of ontology. The objects in an ontology may incorporate solid items like individuals, creatures, tables, autos, atoms and planets, just as unique people like numbers and words (in spite of the fact that there are contrasts of assessment with regards to whether numbers and words are classes or objects). Carefully talking, an ontology needs to exclude any objects; however, one of the overall motivations behind an ontology is to give a method for grouping objects, regardless of whether those objects are not unequivocally a part of the ontology.

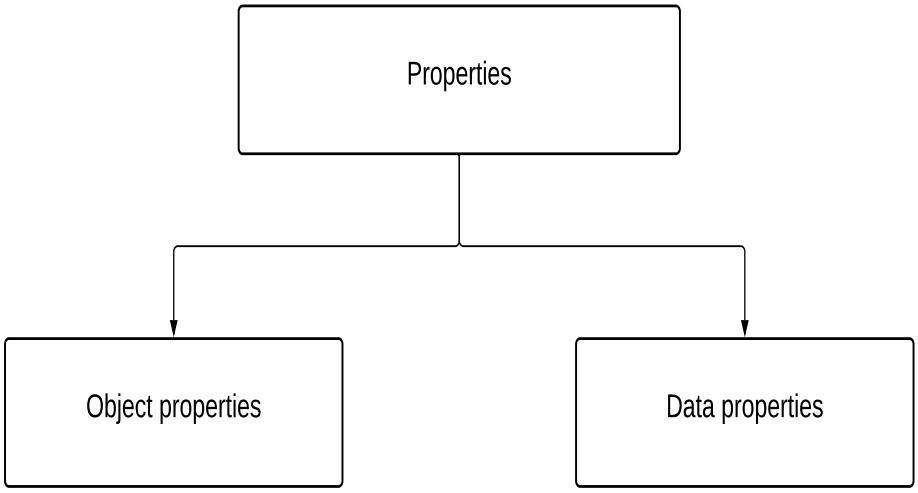


Figure 8. Types of properties

2.3.1.1.3. Properties or Attributes

In an ontology, an attribute is used to define the properties or characteristics of an entity or an object. Generally, there are two types of properties in an ontology, namely the object properties and data properties, as shown in Figure 8. The object properties define the relationship among instances or individuals of different classes, and the data properties describe the relationship among instances and data values.

2.3.1.1.4. Relations

Relationships (otherwise called relations) between objects in an ontology indicate how instances are identified with different instances. Ordinarily, a connection is of a specific kind (or class) that determines in what sense the instance is identified with the other instance in the ontology. A significant kind of connection is the subsumption connection (is-a-superclass-of, the opposite of is-a, will be a-subtype-of or is-a-subclass-of). This characterizes which objects are ordered by which class. The expansion of the is-a-subclass-of relations makes a scientific classification; a tree-like construction (or, all the more, by and large, a part of the way requested set) that unmistakably portrays how objects identify with each other. In such a design, each item is the 'offspring' of a 'parent class'. Another regular sort of relation is the mereology connection, composed as a component of, that addresses how instances consolidate to shape composite instances[73]. Relation types are in some cases area explicit and are then used to store explicit sorts of realities or to address specific kinds of inquiries. Assuming that the meanings of the relation types are remembered for metaphysics, the philosophy characterizes its own ontology definition language. An illustration of an ontology that characterizes its own connection types and recognizes different classifications of connection types is the Gellish ontology.

2.3.2. Constructing an Ontology

2.3.2.1. Determine the Domain and Scope of the Ontology

The development of an ontology begins with determining the domain of the developing ontology and its scope as shown in Figure 9. This is the very first step of constructing an ontology that consists of various tasks, such as to specifically define the ontology domain, defining the purposes of an ontology, the user of an ontology and for what types of questions the information in the ontology should provide answers.

One of the approaches to comprehend the area of ontology is to portray a rundown of inquiries that an information base, depending on the ontology, ought to have the option to reply. This errand is called the competency questions and is valuable to comprehend if the ontology incorporates adequate data or needs further subtleties. The ontology advancement is an iterative interaction, and there are many right approaches to demonstrate a domain; thus, the inquiry rundown ought not to be comprehensive.

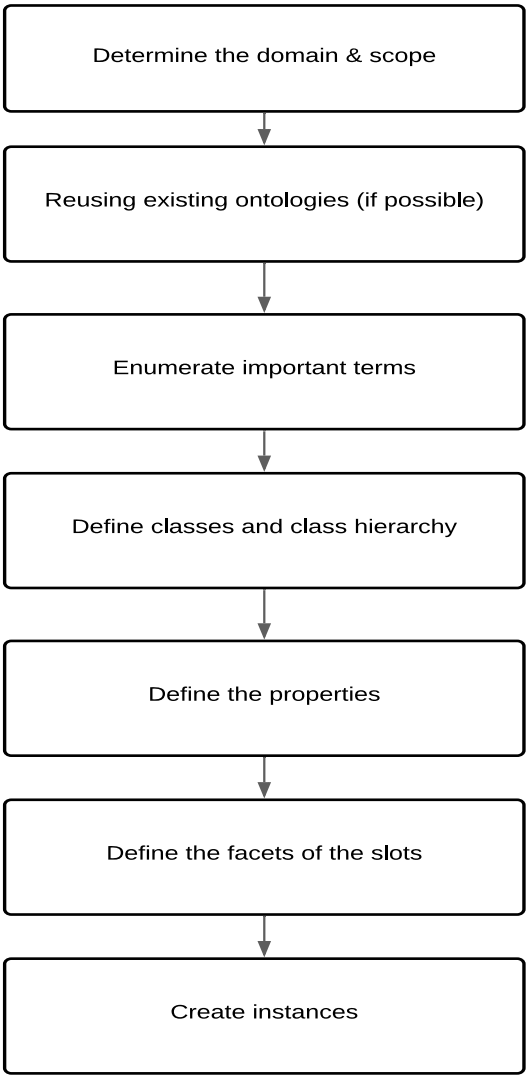


Figure 9. Steps for the construction of an ontology

2.3.2.2. Consider Reusing Existing Ontologies

The second step of building an ontology is the reusing of existing accessible ontologies. The reusing of ontologies is quite often worth thinking about what another person has done and checking on the off chance that can be refined and expand the existing sources for the specific area and assignment. The reusing of a current ontology might be a necessity if the framework needs to connect with different applications that have effectively dedicated to specific ontologies or controlled vocabularies. Numerous ontologies are accessible in electronic shape and

can be brought into an ontology development environment that is being utilized. The formalism where an ontology is communicated frequently does not make any difference, since numerous information portrayal frameworks can import and fare ontologies. Regardless of whether an information portrayal framework cannot work straightforwardly with a specific formalism, the assignment of interpreting ontology, starting with one formalism, then onto the next, is normally not a hard one. Probably the most popular ontologies in the semantic web area are the following [130].

2.3.2.2.1. Dublin Core

This ontology gives a basic and normalized set of shows for portraying things online in manners that make them simpler to discover. Dublin Core is broadly used to depict computerized materials like video, sound, picture, text and composite media like Web pages.

2.3.2.2.2. FOAF

The Friend of a Friend (FOAF) Ontology, characterized by utilizing OWL language, is RDF augmentation, which is made to characterize individuals, their movement and their associations with others and different items. The FOAF project is making a Web of machine-understandable pages, depicting individuals, the connections among them and the things they make and do.

2.3.2.2.3. SKOS

Simple Knowledge Organization System Ontology is a typical information model for sharing and connecting information association frameworks, like thesauri, scientific categorizations, characterization plans and subject heading frameworks, which share a comparable design, by means of the Web.

2.3.2.3. Enumerate Important Terms in the Ontology

After the specification of the ontology domain and defining the scope of the ontology along with considering the reusing of available ontologies, the next step is to enumerate all possible vital terms and their properties of a specific domain for which the ontology is developed. At first, it is essential to get a thorough rundown of terms without stressing over the cover between the concepts they address, relations among the terms, or any properties that the ideas may have, or whether the ideas are classes or slots.

2.3.2.4. Define the Classes and the Class Hierarchy

After gathering the main terms in the third step, the terms addressing concepts, liking to utilize things, and terms addressing relations need to be distinguished. In addition, various levelled connection among the terms need to be characterized, and three methodologies can be utilized. First, a top-down advancement measure begins with the meaning of the broadest concepts in the specific area and then the specialization of concepts.

Second, a bottom-up advancement measure begins with the meaning of the most explicit classes, the leaves of the pecking order with the ensuing gathering of these classes into more broad ideas. Thirdly, a mix improvement measure is a mix of the top-down and bottom-up. The more striking ideas are characterized first and afterwards summed up and practiced properly.

2.3.2.5. Define the Properties of Classes-Slots

In ontology, the classes alone will not provide enough information to answer the competency questions. Once some of the classes are defined, the internal structure of concepts must be described by defining their properties. It must be determined for each property which class it describes. These properties become slots when attached to the classes.

2.3.2.6. Define the Facets of the Slots

A slot can have various facets portraying the type of input, permitted values, the quantity of the qualities (cardinality) and different highlights of the qualities that the slot can take. For instance, the estimation of a "name" space (as in "the name of a wine") is one string. This means that the name is a slot with esteem type String. Some of the common facets of the slots in the ontology are shown in Figure 10.

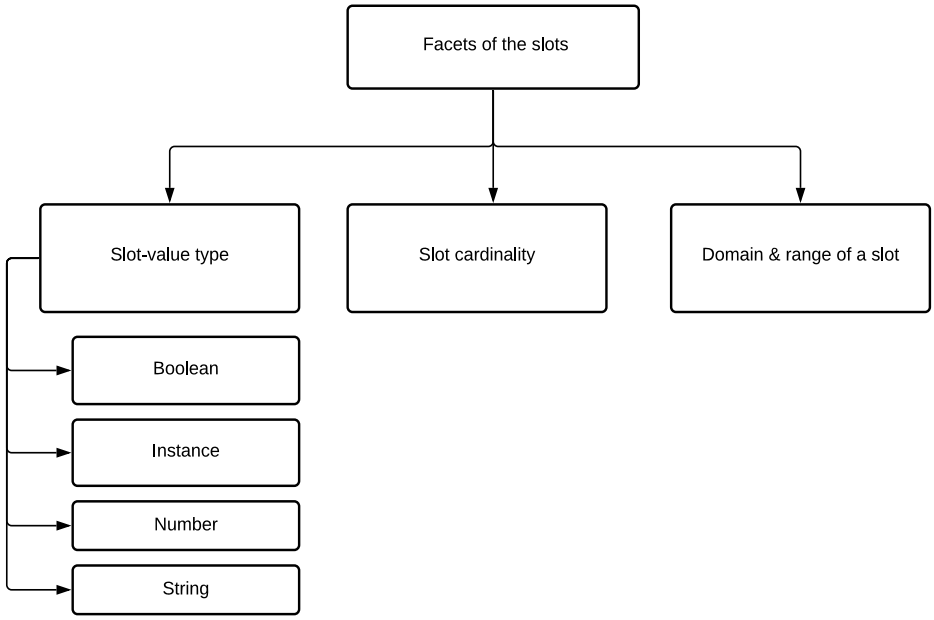


Figure 10. Facets of slots

2.3.2.7. Create Instances

The last action is making the instances of classes in order. In order to define an instance of a particular class, it needs (1) picking a class, (2) making an individual case of that class and (3) filling in the slots qualities.

2.3.3. Literature Studies of Ontologies in Digital Forensics

In this section, numerous ontologies in the digital forensics domain, developed by different authors or researchers from time to time, are presented in ascending order. Brinson, Robinson and Roger [21] stated that the area of cyber forensics is still at the outset with a solid requirement for direction and definition. For this purpose, the authors developed the cyber forensics ontology for identifying the exact layers for specialization, certification and education within the cyber forensics domain. The ontological model split the topic of cyber forensics into two major subtopics, i.e., technology and profession. Subtopic technology is divided into hardware and software. The subtopics hardware and software are split further into large scale digital devices, small scale digital devices, computers, storage devices, obscure devices and analysis tools, operating systems and file systems. The subtopic profession is split into law, academia, military and private sector. The law section focuses on the law enforcement, courts and legal aspects of cyber forensics. Academia is split into research and education, while the military category focuses on what cyber forensics duties are performed by the military personnel. The military section can be defensive and offensive. The private sector was divided into consulting and industry. The objective of creating this ontological model was to define the right levels of education, certification and specialization. The limitation of this approach is that it does not consider and analyse the factors that influence the implementation of this ontological approach [17].

Harrill and Mislan [64] stated that Small Scale Digital Device Forensics (SSDDF) is a new area of study, which needs direction. The small scale digital devices are split into three subparts, including cellular telephones, Personal Digital Assistants (PDA) and software components. The devices and their corresponding forensics processes are not transparent. The objective of this paper was to design an ontological approach to provide law enforcement with appropriate knowledge about the devices that were found in the SSDD domain. The limitation of this ontological approach is that it is not able to address both old-fashioned as well as the latest digital devices [17].

Kahvedzic and Kechadi [75] propose DIALOG, i.e., Digital Investigation Ontology, a model to encapsulate the knowledge related to digital investigation cases. DIALOG presents a dictionary of concepts and associations in the form of an ontology to define the semantics of cases. In order to define the four areas of the case, four sub-ontologies are defined, namely cybercrime type, types of data locations, type of data itself and the tools used to find that data. The purpose of the model is to describe the properties and attributes of vital forensics ideas for better understanding of investigators. The authors demonstrated the implementation of

DIALOG ontology by modelling the semantics of the knowledge, associated with the Windows Registry. The major limitation is that the authors considered only a single-source, i.e., the registry, to annotate evidence and that is not sufficient to demonstrate the capability of the approach [17].

Park, Cho and Kwon [104] developed Cyber Forensics Ontology for the cyber investigations in cyberspace. The authors classified Cybercrime into two classes, i.e., cyberterror and general cybercrime, and how these two classes are related. The analysis of cyberterror needs the latest technology, system environment and experienced experts, and general cybercrime is related to a general crime by digital evidence. The authors described the concepts and associations between crime types, evidence collection, criminals and criminal case and law. The drawback of this approach is that it is least based on the digital evidence and other stages that are essential to collect and interpret the digital evidence in the digital investigation process. Only one stage, namely "collection", is mentioned by the authors [17].

Saad and Traore [117] describe the concept of an intelligent network forensics system that reconstructs intrusion scenarios and makes attack attributions that require knowledge about intrusions signatures, evidence, impacts and objectives. They as well have mentioned that recently, several machine learning techniques have been proposed to automate and develop intelligent network forensics systems, but they are not able to achieve it. For this, they proposed an ontology-based network forensics knowledge representation approach. This ontology gives a formal description of the concepts, defining the network forensics domain, and describes the associations. The developed ontology is based on the hyper approach that combines different major features from distinct ontologies and is known as METHONTOLOGY. The developed ontology consists of 111 classes, four taxonomic relations and 21 ontological relations. The limitation of this approach is that it is complicated and required a lot of time and effort to compose the ontology.

Ćosić, Ćosić and Bača [41] highlight the concept of chain of custody of digital evidence and its importance in the digital forensic field. They as well focus on factors that can affect the chain of custody of digital evidence, such as who, when, where, why, and how, during various stages of digital investigation process. In order to handle the chain of custody of digital evidence, the authors defined taxonomy and developed an ontological approach that is known as DCoDeOn (Digital Chain of Custody Digital Evidence Ontology) by using the Protégé tool. The developed ontology consists of five major modules, namely characteristics, dynamics, factors, institutions and integrity. The taxonomy was based on a top-down approach, i.e., the most specific concepts were first defined and specialized afterwards and designed in such a way to simply insert new class, slots, property constraints and common facets. [17].

Ćosić and Ćosić [40] highlight the problems encountered by the digital practitioner during the digital investigations to interpret digital evidence, primarily because of misinterpreting or false understanding of various vital concepts [129]. An ontology of digital evidence was developed to figure out this problem. According to the authors, this ontology can be used to share a common understanding of the

structure of this domain (digital forensics) between forensics investigators and other personnel that have to deal with digital evidence, among software agents and between forensics investigators and software. This ontology was only explained theoretically, which is a limitation of the research [17].

Luthfi [85] highlights the importance of the retrieval of evidence in digital forensics. The author proposed an ontology framework approach for the extraction of evidence. This approach is based on the hierarchy of layers and contains two layers that are called analysis tools and operating systems. The second layer is automatically developed to assist the practitioner in retrieving the evidence by using rule-based analysis (forward chaining). The proposed ontology framework consists of two layers in a hierarchical structure, namely hardware and software. The hardware layer is further divided into two sub-layers, i.e., a large scale digital device (LSSD) and a small scale digital device (SSSD), and the software layer is as well further divided into sub-layers, i.e., the analysis tools and operating systems.

Chabot et al. [32] stated that digital forensics investigators face challenges, such as the high volume of data [1], which are becoming continuously vast and diverse because of the growth of new technologies. Therefore, the interpretation of digital evidence and the reconstruction of events is a complicated and time-consuming task for the investigator. Moreover, the authors as well identified seven vital factors that a reconstruction tool must have to handle these three challenges, namely volume, heterogeneity and legal requirements. An approach to SADFC, i.e., Semantic Analysis of Digital Forensics Case, based on three layers of ontology, called ORD2I, is introduced to present any digital events. This ontology is related to a collection of tools for obtaining information from the sources of evidence, instantiating ontology, inferring new knowledge and interpreting it. However, the performance of the approach still needs to improve [17].

Alzaabi et al. [6] highlighted the issues of smartphones that the digital investigator faces during the analysis of data, such as volume and complexity of data. In order to handle these issues and support the investigator, an automated technique, called Forensic-Driven Ontologies for Smartphones (F-DOS), is developed to quickly and efficiently analyse the data. The proposed methodology is based on a modified version of an ontology development methodology called Methontology. The development of F-DOS ontology begins with identifying the general purpose and scope of ontology, requirements of ontology (competency questions), reusing of ontologies (if possible), collecting concepts and relationships among them, evaluating ontology and formalization of ontology. The F-DOS ontology consists of two main ontologies, namely upper ontology and domain ontology. The domain ontology is further divided into four sub ontologies, namely contact ontology, message ontology, investigation case ontology and other domain ontology. The content of the smartphones can be extracted by using the knowledge base of F-DOS with the help of a standard query language called SPARQL. The major drawback of this work is that this approach is specified only theoretically.

Karie and Kebande [80] stated that digital forensics is a relatively new discipline with various technical and non-technical terminologies that can be hard to

comprehend. The main problem addressed by the authors is that there is no approach in digital forensics that can help investigators in reasoning concerning the perceived meaning of different digital forensics terminologies that were encountered during the digital forensics investigation process. In order to solve this problem, the authors examined the concept of developing ontologies for digital forensics terminologies and proposed an ontological approach to resolve the meaning of different digital forensics terminologies. Moreover, the approach was only discussed theoretically and is a drawback of this paper [17].

Kalemi and Yayilgann [77] focus on the benefits of online social networks (OSNs) in digital forensics. The authors stated that no ontology uses OSNs data to support the investigation process. For this, the author proposed an ontology called SC-Ont. The SC-Ont ontology consists of three pillars, namely people, crime and crime-solving. The main objective of the proposed ontology is to provide an ontological prototype for supporting crime-solving by using data found in OSNs. Moreover, this ontology is not yet fully developed and implemented [17].

Wimmer, Chen and Narock [133] stated that technology evolves at a rapid speed. Thus, the digital forensic field needs to be continually adapting by developing novel tools and techniques to implement forensic analysis on many different systems, such as desktops, portable devices, sensor devices and many more [67]. The authors mentioned that the researchers use the concept of ontology to classify the digital forensics.

Table 3. Available ontologies in digital forensics domain

Ontologies in digital forensics domain	Achievements	Problems
Brinson, Robinson and Roger [21]	Developed cyber forensics ontology, highlighted the profession of cyber forensics that can help in curriculum development.	Do not evaluate the factors that influence the implementation of this approach and the lack of advanced cyber forensics methodologies.
Harrill and Mislan [64]	Developed a framework to place small scale digital devices.	Do not consist of all versions of small scale digital devices.
Kahvedzic and Kechadi [75]	Developed digital investigation ontology (DIALOG), a model to encapsulate the knowledge related to digital investigation cases.	Encoding of forensics information is related only to Windows registry source.

Park, Cho and Kwon [104]	Developed the ontology to describe the crime types and the relationship among them.	The proposed ontology is least based on the digital evidence and other stages that are essential to collect and interpret digital evidence.
Saad and Traore [117]	An ontology-based network forensics knowledge representation approach is proposed to describe the concepts of the network forensics domain and their relationships.	The proposed ontology is complicated and required a lot of time and effort.
Ćosić, Ćosić and Bača [41]	Digital Chain of Custody Digital Evidence Ontology (DCoDeOn) is developed by using the top-down approach in such a way that allows insertion of new classes and properties.	The proposed ontology is not technically evaluated.
Luthfi [85]	An ontology framework is developed to acquire digital evidence automatically.	The evaluation of ontology framework is not discussed.
Chabot et al. [32]	The ontology ORD2I (Ontology for the Representation of Digital Incidents and Investigations) is developed to show any digital incident and associated digital investigation.	The performance of a proposed approach is slow and handles limited sources of information. Moreover, the developed ontology is not evaluated.
Alzaabi et al. [6]	Forensic-Driven Ontologies for Smartphones (F-DOS) are developed to quickly and efficiently analyse the data.	The developed ontology is not evaluated and discussed only theoretically.

Karie and Kebande [80]	An ontological approach is developed to resolve the meaning of different digital forensic terminologies.	The proposed approach is only theoretically discussed, and no ontology evaluation is performed.
Kalemi and Yayilgann [77]	An SC-Ont ontology is developed to use the information from online social networks (OSNs) in digital forensics.	The proposed ontology is incomplete.
Wimmer, Chen and Narock [133]	An ontology is developed to allow digital investigators to choose appropriate forensics tools for the investigation of a digital crime.	The verification and validation of developed ontology are not performed.

Still, there is no ontology to define the capabilities and relationships among various digital forensic tools. In order to address this issue, they developed an ontological approach based on Resource Description Framework (RDF) and Web Ontology Language (OWL), which is searchable by using SPARQL and a list of standard digital forensic tools. The main objective of ontology is to assist digital investigator in selecting the appropriate tools for the analysis of digital devices [17].

Akremi et al. [3] stated that the big data produced by the web services make the investigation process complicated and time-consuming. For this, the authors proposed an extensible standards-based semantic ontology for the representation of web service log data. The main aim of the ontology is to extract hidden information and eventual scenarios of cyber-attacks in weblogs. Digital investigators can specify validation rules and execute them by using a logical reasoner over the proposed ontology to get a forensic report. According to the authors, the proposed ontology can support the investigator in analysing the task and minimize the required time [17].

Based on the literature studies (see Table 3), it has been found that the digital investigators encountered various technical and non-technical terminologies that can be hard to comprehend and faced challenges in interpreting digital evidence, primarily due to the misunderstanding of certain vital terms. Various approaches are developed to assist digital practitioners in understanding newly encountered terminologies, each one with its drawbacks and not being able to assist investigators in reasoning about the perceived meaning of different encountered digital forensics terminologies. Moreover, the literature studies have shown that the existing ontologies are not technically verified and validated.

2.3.4. Ontology Evaluation

The evaluation of ontology can be described as "a technical judgment of the content of the ontology with respect to a frame of reference during every phase and between phases of their lifecycle" [61]. A frame of reference consists of a specification of requirements, competency questions, real-world and many more. The evaluation of ontology consists of two parts, namely verification and validation.

The evaluation of ontology has increased as well due to the common use of ontology. For this, numerous distinct evaluation methods have been developed by different researchers and authors. A few authors recommend manual and others suggest the automatic approach. There are some methods for handling the taxonomy of ontologies, and few methods evaluate the content of ontologies. Some methods are dependent on a particular tool or language, while some methods can be utilized independently from a tool or language. As a wide variety of methods are available for the evaluation of ontology, one needs to be careful while selecting the methods that will contain both aspects of evaluation, namely verification and validation.

2.3.4.1. Ontology Verification

The verification of ontology can be defined as an accurate depiction of a domain and the need to check its definitions. It means that the hierarchy of concepts needs to be consistent and as well right according to the real world. For this purpose, an independent method, namely ontology taxonomy evaluation, is available [59 & 79]. This method is used manually, and the ontology is verified according to three main factors, namely inconsistency, incompleteness and redundancy.

The consistency can be de defined as the ontology that does not contain or allow any contradictions. A given definition of the ontology is consistent if and only if the individual definition is consistent and no contradictory sentences can be contained from the other definitions. The first factor, namely inconsistency, consists of detecting three types of errors in the ontology, i.e., circularity errors, partition errors and semantic errors. The first error, i.e., circularity errors, exists in the otology when a class is described as a specialization or generalization of itself. Partitions can be described as defined concept classifications in a disjoint and/or complete manner. Partitions errors occur if the ontology contains an incorrect description of disjoint classes, such as a class is a subclass of two or more disjoint classes and an instance is an object of two or more disjoint classes. The third error, i.e., semantic errors, occurs when ontology contains inaccurate semantic classification, which means that it classifies a concept as a subclass of a concept, to which it does not actually belong.

Incompleteness is one of the primary problems in ontologies. An ontology is incomplete if the definition of one class or concept is missing in the established reference framework. The ontology is complete if all that is needed to be in the ontology is explicitly stated in it and every definition of ontology is complete. The second factor, namely incompleteness, consists of detecting two types of errors in the ontology, namely incomplete concept classification and partition errors.

Incomplete concept classification errors occur when some concepts of a domain are absent from the taxonomy of ontology. The second error, i.e., partition error, occurs when the relations between some classes or concepts are not defined in the ontology.

The redundancy error occurs when an expression of the ontology is redefined that is already described explicitly or that can be inferred from the other definitions. The third factor, namely redundancy, consists of detecting three types of errors, namely grammatical redundancy, identical formal definition of some classes and identical formal definitions of some instances. The grammatical redundancy error occurs when more than one definition of a class or an instance is given or defined. The second error, i.e., identical formal definition of some classes, occurs when two or more classes of the ontology have the same definition, and there is only one difference between the classes, i.e., the names of the classes. The last and third errors, i.e., identical formal definitions of some instances, occur when two or more instances of the ontology have the same definition, and there is only one difference between instances, i.e., the names of the instances.

2.3.4.2. Ontology Validation

The validation of ontology can be described as whether the definition of ontology really models the real world for which the ontology is developed. The main aim of ontology validation is to prove that the world model is compliant with the world modelled formally. Since the validation of ontology refers to the real world; thus, the evaluation of the content of the ontology is required [76]. For this, an independent method, namely ontology content evaluation, is available. This method is performed manually, and the ontology is checked according to five factors, namely consistency, completeness, conciseness, expandability and sensitiveness. The first two factors, namely consistency and completeness, are already described in the section above, i.e., "ontology verification". The third factor, i.e., conciseness, the ontology is concise when it does not contain any irrelevant definitions of classes or concepts along with redundancies of definitions. The fourth factor, i.e., expandability: the ontology is expandable if there is no need to change a set of defined definitions while there is a need to add new definitions to the existing ones. The last and fifth factor, i.e., sensitiveness, the ontology is not sensitive if small changes in the definition do not change a set of well-described concepts.

2.4. Visualization in Digital Forensics

Despite the growing popularity of visualization [86], it is not simple to define visualization in such a manner that would correspond to the entire categories of visualization strategy that is being constructed nowadays and at the same would differentiate the visualization from other similar fields, namely scientific visualization and information design. In the [82], the authors define: "visualization is the communication of abstract data through the use of interactive visual interfaces". In the [109], the authors define: "visualization utilizes computer graphics and interaction to assist humans in solving problems." The evolution of modern electronic devices is outpacing the scalability and effectiveness of the tools

that are used to analyse the digital evidence recovered from them [14, 27]. Indeed, current digital forensic techniques and tools are unable to handle large datasets efficiently. As a result, the time and effort required to conduct digital forensic investigations are increasing [101 & 28]. The two key challenges that the digital forensic investigations are facing are the complexity and volume of digital evidence. The complexity arises from the heterogeneous and idiosyncratic nature of digital evidence [105, 30]; the evidentiary data is expanding over digital devices, every device with its own methods for saving and displaying the data. Moreover, the number of digital evidence grows as the cost of storage devices becomes reasonable and increasingly huge, and high-speed processor and connectivity of the internet with high-bandwidth nowadays facing digital devices to be used all the time and everywhere. The implementation of information visualization techniques can handle the heterogeneity and huge quantity of information or digital evidence. In the paper [101], the authors enhance the digital forensic investigation process by integrating information visualization techniques into the existing digital forensic investigation workflow that is known as Explore, Investigate and Correlate (EPIC) process. The EPIC process builds on the "visual form" components of the visualization reference model as shown in Figure 11 [36]. In the paper [36], the authors stated that the visualization process is an exploration process. Given a data set Cdata, it needs to take decisions related to which visualization tools a user wants to use for searching the data set. After that, the user uses and tries distinct controls (Cctrl), which

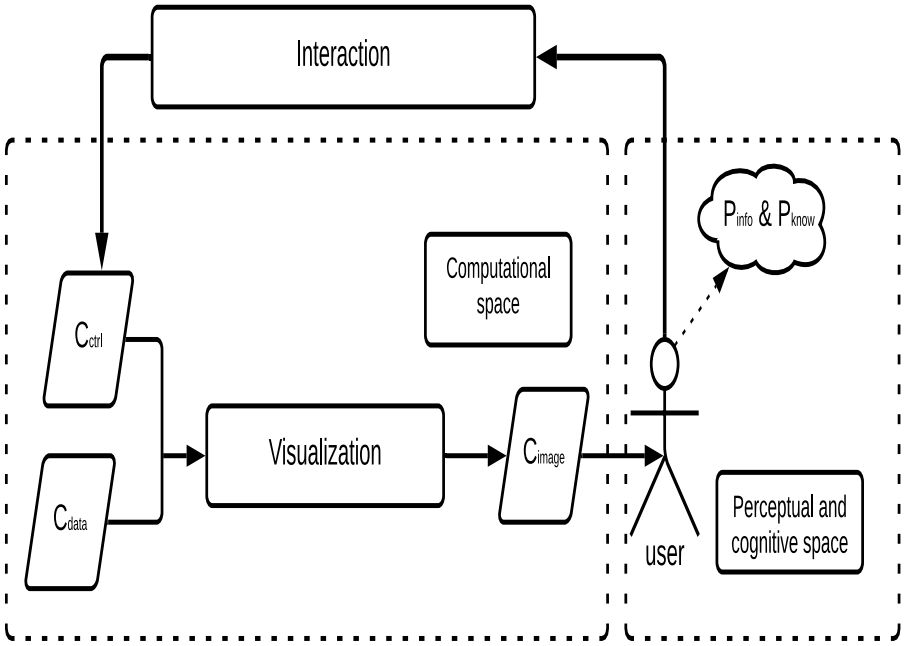


Figure 11. Visualization as a search process [36]

includes styles, layout, viewing position, colour maps, transfer functions and many more till the user gets a satisfactory selection of visualization output. It relies upon the tasks of visualization; the satisfaction can come in various forms. For instance, a user may have adequate information about the particular data set or may have the most convenient presentation of the information to help others in acquiring knowledge. The aim of visualization is to display a figure to the user in such a way that allows the user to interpret the underlying information [34]. The visual data are input of the visualization that needs to be presented and the control data given by the human interface. There are two spaces in the visualization as described in the search process model proposed by [35]: firstly, "computational space," where the visualization is updated, controlled and novel visions are created; secondly, "perceptual and cognitive space," where the users can view a novel image and collect information and knowledge related to the data being visualized and searched. Outside of these spaces, there is human interaction with the system of visualization, which oversight the visualization and the creation of new views. The output of the EPIC process visualization framework can illustrate the digital evidence simply and intuitively, enhancing decision making and facilitating the explanation of phenomena in evidentiary data.

Table 4. Visualization techniques in digital forensics domain

Information visualization techniques	**Strengths**	**Drawbacks/Problems**
Geometric transformed displays visualization [34, 137 & 131]	Capable to display multidimensional data.	Not suitable for presenting hierarchical relation and textual information.
	Geometric techniques: Prosection views, Hyperslice, Scatterplot-Matrices, Landscapes, Projection pursuit and Parallel coordinates techniques.	
Iconic displays visualization [34, 17 & 114]	To facilitate the visualization of data by using features of icons.	Unable to visualize textual data or information.
	Iconic techniques: Chernoff-Faces, Stick figures, Shape coding, colour icons and Tilebars.	

Pixel displays visualization [116 & 95]	To map the attribute value of the data to a single coloured pixel and show the most possible information at a time.	Unable to illustrate the distribution of data in a multidimensional space.
	Pixel techniques: Recursive pattern, Circle segments and much more.	
Stacked displays visualization [65 & 139]	To visualize information by using hierarchical partitioning into sub spaces.	Fails to visualize textual information.
	Stacked techniques: Dimensional stacking, Worlds-within-Worlds, Tree map and many more.	
Graph based visualization [48]	To visualize large graphs by using techniques to convey the meaning of graphs and capable to visualize large data set and relationships.	Complexity and readability.
Traditional visualization techniques [48]	These techniques compose information that is visually interesting to the audience and can directly emphasize the key findings.	A disadvantage of using a chart is that, by design, a chart will likely not be as precise as the raw data and not suitable for a very huge volume of data.
	Traditional visualization techniques: Line charts, Pie charts, Bar charts, Area charts and many more.	
Visualizing big data techniques [48]	These techniques can visualize the huge volume of data along with the consideration of challenges, such as volume, velocity and	Complicated to interpret visualization.

	variety of data.	
	Visualizing big data techniques: word clouds, symbol maps, connectivity charts.	

A geometrically transformed display visualization technique is used for multidimensional data. It finds the interesting transform in multidimensional data. These techniques include exploratory statistics, such as scarred plot matrices [137], and techniques, which are subsumed under the projection pursuit term [100]. Different geometric transformed techniques are the prosection view technique, hyperslice technique and parallel coordinate visualization technique [81]. In the parallel coordinate technique mapping of the k-dimensional space onto the two display dimensions is done by using k equidistant axes, which are parallel to the display axes. These axes correspond to the dimensions linearly scaled from the minimum to the maximum value of the corresponding dimension [131]. A data item is visualized as a polygonal line, intersecting each of the axes at that point of the considered dimensions. In this technique, useful patterns were discovered either by using association rule mining or the decision tree method. These data mining techniques are used in geometrically transfer display visualization [34].

Iconic displays visualization techniques are as well used for the visualization or exploration of multimedia data. Icons can be arbitrarily defined as a little face, TileBars, star icons, colour icons, stick figure icons [17] and needle icons as used in MGV. They are put in the middle of columns. By mapping the attribute value of data record in a data set with the feature of icon data visualization is done. In the stick figure icon technique, the display dimensions are mapped to two dimensions, and the rest of the dimensions are mapped to the limb length of the stick figure icon [114]. The angle of stick icon is considered as well for mapping. The patterns vary with respect to the characteristics of data. If the data items are large in size with respect to the two display dimensions, the resulting visualization presents texture patterns. Therefore, the varying patterns are detectable by preattentive perception [105].

In dense pixel display technique map, each dimension value belongs to a coloured pixel value, and group pixels belong to each dimension of a specific area [116]. This technique uses one pixel per data value. A generally large amount of data visualization is possible by using a dense pixel display. Pixel represents data value; thus, the arrangement of all pixels in display is adjusted according to the purpose. Therefore, different purpose has a different arrangement of visualization. By arranging pixels in a proper manner, the resulting visualization provides detailed information about the data. The most commonly used example is recursive pattern and circle segment. According to the data attribute natural order of the database, the arrangement is the aim of recursive pattern techniques. A user may specify the parameter and control arrangement of the pixel for each recursion level, which leads

to a meaningful substructure. Back and forth arrangement is done on each recursive level with height and width. The width is provided by the user. In the circle segment technique, the data in the circle are divided into segments for each attribute. The pixel arrangements start from the centre and continue outside. All attributes are closed to the centre; thus, the data are displayed orthogonally [95].

The stacked display technique is used to present hierarchical partitioning. The data dimension is used for partitioning the data, and appropriately, there is a select hierarchy. For example, dimensional stacking, in this technique, one coordinate system is inside another coordinate system. If the outmost level coordinate is divided into the rectangular cell and within the cell, and the next two attributes are used span to display the information on the second coordinate system [65]. The usefulness of the visualization mostly depends on the distribution of outer coordinate data. Therefore, the selection of the outer points dimension is the most important. For that, the thumb rule is used for the selection of dimensions [139].

A graph is a collection of nodes (or vertices) and edges (or links). Each node represents a single data point (a person, a phone number, a transaction), and each edge represents how two nodes are connected (a person possesses a phone number, for example). This way of representing data is well suited for scenarios involving connections (social networks, telecommunication networks, protein interactions and a lot more) [139]. The graph visualization is the visual representation of the nodes and edges of a graph. The dedicated algorithms, called layouts, calculate the node positions and display the data on two (sometimes three) dimensional spaces. There are numerous benefits of using graph-based visualization techniques, such as a higher chance to discover insights, better understanding of a problem, an effective form of communication, and anyone can work with visualization [48 & 125].

There are numerous visualizations techniques (see Table 4) that are available, which can be used for visualizing the information or data. For instance, x-y plots, bar charts, line graphs and many more. Generally, the visualization technique depends on a number of dimensions and variables available in the data. According to variables and dimensions, the data can be divided into one-dimensional data, two-dimensional data, multidimensional data and more complex form text and hypertext data, the hierarchy of graph, data type from the field of algorithm and software. In order to illustrate one-dimensional data, histogram or pie chart method is used; for two-dimensional data, scatter plot and line graph is used; for multidimensional data, icon-based method, pixel-based method, dynamic parallel coordinate system; for text data graph, chart or network is used. No single visualization technique is best all the time; the performance of a visualization technique is highly dependent on the data. Moreover, Table 4 shows the strengths and drawbacks of distinct visualization techniques that are available in digital forensics domain.

2.5. Summary

Digital forensics can be described as a series of discrete activities performed by different specialists of the digital forensics domain to understand information and attain digital evidence from digital devices. As there are numerous reasons for

performing digital investigations, some of them are to attain, interpret the digital evidence and present it in the court, to identify a leak within an organisation and many more. Nowadays, digital forensics investigation is not only limited to the retrieval of data and digital evidence from the computer system, it includes other digital devices such as smartphones, which are extensively used for communication, accessing the information on the web and much more. The digital forensics domain as well faces various challenges because of the explosive growth of the internet, rapid innovation in technologies and continued growth in the usage of digital devices. Some of the primary challenges include huge volume of data, heterogeneity of data, visualization of digital evidence, presentation and validation of digital evidence in the court and many more.

The construction and investigation of the timeline is a great approach to identify numerous activities that take place on a particular digital device and attain digital evidence. The literature studies have shown that there are numerous existing approaches for the reconstruction of a timeline to interpret the activities performed by the user. Moreover, a comparative study is performed among numerous available timeline reconstruction approaches based on five major key issues in a digital forensics investigation that should be handled in order to understand the timeline precisely and find the digital evidence. These five major issues include: heterogeneity and huge volume of sources of data, automatic extraction of events and knowledge from the timeline, a clearly defined investigation model that allows describing the process used to get the required results, analysis capabilities and the integrity of data. The comparative studies show that most of the existing approaches are capable to handle and fulfil the automatic extraction of the timeline and volume and the heterogeneity of data issues. The other key issues, i.e., theory and integrity of data, are fulfilled by very few existing approaches. However, the analysis of timeline issue, i.e., to assist the digital investigator or user in understanding the timeline, composing timeline easy to read, recognizing the correlation between events, producing a conclusion in the form of digital evidence from the knowledge contained in the timeline, is not addressed and fulfilled by the available approaches. For this, a novel approach based on the abstraction concept is developed to analyse the timeline and assist the digital investigator or user in interpreting the timeline, and it is discussed in detail in chapter 3.

There are different reasons for developing ontology of any particular domain, such as common understanding, reusing, sharing, analysing of the domain knowledge, separation of domain knowledge from the operational knowledge and making domain assumptions explicitly. It can be summarized that ontology is the representation of knowledge of a domain, where a set of objects and their relationships are described by a vocabulary. Moreover, a typical ontology of a particular domain is comprised of different components, such as classes, instances, properties and relations. The development of ontology for a required domain follows various steps, starting with the determination of the domain and scope of the ontology, reusing of ontologies, considering paramount terms, defining classes and their hierarchy, defining properties of classes, and ending with the creation of

instances of classes. The literature studies as well show that the evaluation of ontology has increased due to the frequent usage of ontology. In short, the evaluation of ontology can be described as a technical judgement of the ontology and consists of two parts, namely ontology verification and ontology validation. Ontology verification refers to the correct illustration of the definition and hierarchy of a domain, according to the real world. However, ontology validation refers to the evaluation of the content of the ontology and is compliant with the world model.

Various ontologies are developed by different authors and researchers in the domain of digital forensics. The most known are cyber forensics ontology, Digital Investigation Ontology (DIALOG), Digital Evidence Semantic Ontology (DESO) and Forensic-Driven Ontologies (F-DOS). However, all of them are goal-oriented, and they are developed for specific goals and objectives in specific cases or scenarios. Since these ontologies cannot be applied for the different cases or scenarios, the existing ontologies of the digital forensics domain are as well not able to assist digital investigator in interpreting new terminologies that are encountered during the analysis of the timeline. Therefore, the author has developed a novel ontology backed by the abstraction approach for the analysis of the timeline, and it is discussed in detail in the following chapter 3.

Visualization is a process of utilization of computer graphics to assist humans in solving their problems. The visualization techniques can be integrated into the process of digital investigation to address the problems faced by the digital investigators, such as complexity, heterogeneity, the volume of digital evidence and many more. The process of integration of visualization technique into the digital forensic process is known as explore, investigate and correlate (EPIC) process. The purpose of visualization is to output an image to a user in a manner that facilitates the understanding of the underlying information. The literature studies show that there are various different visualization techniques available in the digital forensics domain, namely geometric transformed displays visualization, iconic displays visualization, pixel displays visualization, stacked displays visualization and graph-based visualization. The selection of visualization techniques is highly dependent on data, as no visualization technique is suitable for all types of data. In this research work, the outcome of the abstraction based approach for the analysis of the timeline is in the form of textual data or information. Thus, the graph-based visualization technique is used for visualizing the outcome of the abstraction based approach and is discussed in chapter 4.

3. RESEARCH DESIGN AND METHODS

3.1. Overview

This section presents a detailed description of the abstraction based proposed methodology for the analysis of timeline and a proposed novel ontology backed by the abstraction approach to define distinct encountered terminologies during the analysis of the timeline for different operating systems based devices. Moreover, this section contains two major sections and six sub-sections.

- In section "The proposed methodology for analysis of the timeline", the idea of a novel approach for the analysis of the timeline is presented, and it consists of three sub-sections.
- "The novel ontology based on the proposed methodology" section is devoted to presenting the novel ontology to define various encountered terminologies and relationships among them. This section is further divided into three sub-sections.

3.2. Proposed Methodology for the Analysis of Timeline

Different limitations and issues in the existing approaches for timeline reconstruction are featured in the literature studies segment; they come from an immense volume of information when the timeline is extricated from a disk image file, especially with the 'Super Timeline' approach. The exploration work depends on the information given by the command-based digital forensics tools, i.e., Log2timeline and Psort in the form of plaso file, as shown in [3]Figure 12. Additionally, Psort allows converting the plaso file into common file formats, such as L2TCSV, i.e., Log2timeline Comma Separated Values. The L2TCSV timeline contains 17 fixed fields as listed in Table 5; then, the data from the L2TCSV file can be imported into an Excel sheet.

The Log2timeline and Psort digital forensics tools generate a well-structured timeline; however, the immense volume of the data, where all the events are presented, even the smallest ones, overshadow important deciding events that have a big influence. The structured timeline needs to be transformed to collect digital evidence by analysing the timeline. The generated structured timeline consists of many issues; some of them include the repetition of data presented from different

[3] *Some passages have been quoted verbatim from the following source:*

An ontology based on the timeline of Log2timeline and Psort using abstraction approach in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

resources, the volume of data, the heterogeneity of data and many more. Thus, all these issues compose a complex timeline and restrain the digital investigators to understand the timeline and identify different types of activities performed by the user on a particular digital device. The investigators are lost in a huge amount of events.

In order to address these issues and transform a structured timeline, a novel approach based on the abstraction concept is developed by analysing the structured timeline generated by the digital forensics tools, namely Log2timeline and Psort. First, the structured timeline is analysed to identify different kinds of activities or [4]events and artefacts and their relevance performed by the user on a particular digital device.

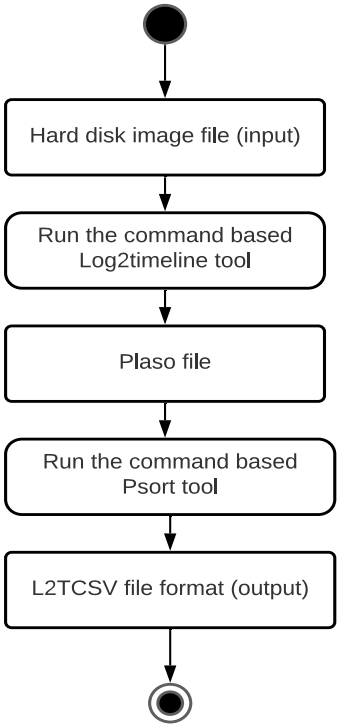


Figure 12. Generation of timeline

[4] *Some passages have been quoted verbatim from the following source:*

An abstraction based approach for reconstruction of timeline in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

In digital forensics, an event can be characterised as an activity performed by the users utilising the digital devices. However, “artefact” currently does not have a formal definition within the domain of cyber/digital forensics, resulting in a lack of standardised reporting and linguistic understanding between professionals. Generally, the artefact can be defined as a piece of data that may or may not be relevant to the investigation/response. The examples of artefacts include registry keys, files, timestamps and event logs. In other words, the artefacts can be defined as something observed in a scientific investigation or experiment that is not naturally present but occurs as a result of the preparative or investigative procedure [16].

The analysis of the timeline of events and artefacts of a particular computer system assists in reducing the complexity of the timeline by splitting it into different and relevant levels of the timeline of events and artefacts. The splitting of the timeline is based on a number of events and artefacts that are available in the timeline along with their significance or importance and the level of detail (abstraction) of information of the timeline. Moreover, it has been found that there are some primary and secondary events in the timeline, which are required to be understood by the digital practitioners to retrieve digital evidence. An event can be primary (main) or secondary (auxiliary); it depends on the relevance or importance and level (high or low) of impact of the event on the timeline of the digital device. Moreover, the repetition of the same time unit numerous times is as well one of the major issues of a structured timeline, which cause the timeline to become complex and irrelevant.

Table 5. Fields in the L2TCSV File by Psort tool [16]

Field	Description
Date	date when the event occurred
Time	time when the event occurred
Timezone	timezone that was used to call the tool with
MACB	Modification, Access, Creation and Birth
Source	short name of the source, such as registry entries are REG
Source type	description of the source
Type	timestamp type, such as last accessed or last written
User	what username is associated with event if any
Host	what hostname is associated with entry if there is one

Short	this contains a short description field where text is stored
Desc	this is where the majority of information that is parsed is stored
Version	gives the version number of the timestamp
Inode	gives the inode number of the file being parsed
Notes	additional storage location for the information for some input modules
Format	input module, which was used to parse
Extra	parsed information that are joined together and stored here; all these pieces of information make up the super timeline that Log2Timeline creates

For example, time unit "11:49:53" in the field "time" is rehashed on numerous occasions, corresponding to different sources of information, for example, "WEBHIST" and "LNK" in the field "source", which shows that the event of a similar occasion has brought a reflection in various logs and various antiquities, as demonstrated in Table 6. Such situations compose a huge, perplexing and hard to break down timeline for digital forensics professionals. Along with these lines, the author concentrates on such situations from the timeline and supplants utilizing appropriate strategies (for example, one occasion for every time unit to kill duplication) to recreate a simple, minimal, conspicuous and organized timeline of events for the investigators.

Table 6. Timeline with repetition (duplicity) of the same time unit [16]

Date	Time	MACB	Source	Source type	Type
19 November 2017	11:49:53	.A..	WEBHIST	Firefox History	Last Visited Time
19 November 2017	11:49:53	.A.B	LNK	Windows Shortcut	Creation Time; Last Access Time
19 November 2017	11:49:53	MA.B	LNK	Windows Shortcut	Content Modification Time; Creation Time; Last Access Time
19	11:49:53	.A.B	LNK	Windows	Creation Time; Last

November 2017				Shortcut	Access Time
19 November 2017	11:49:53	.A..	WEBHIST	Firefox History	Last Visited Time
19 November 2017	11:49:53	MA.B	LNK	Windows Shortcut	Content Modification Time; Creation Time; Last Access Time
19 November 2017	11:49:53	MA.B	LNK	Windows Shortcut	Content Modification Time; Creation Time; Last Access Time

Thus, based on the importance and level of impact of events and artefacts on the timeline of a computer system along with the level of detail (abstraction) of the timeline, the structured timeline is split into four levels of timeline of events and artefacts, namely Events: high level (new entries and web surfing), Events: low level (web surfing, actions of modifying), Artefact location: high level (include all application files) and Artefact location: low level, as shown in Figure 13 [16]. The

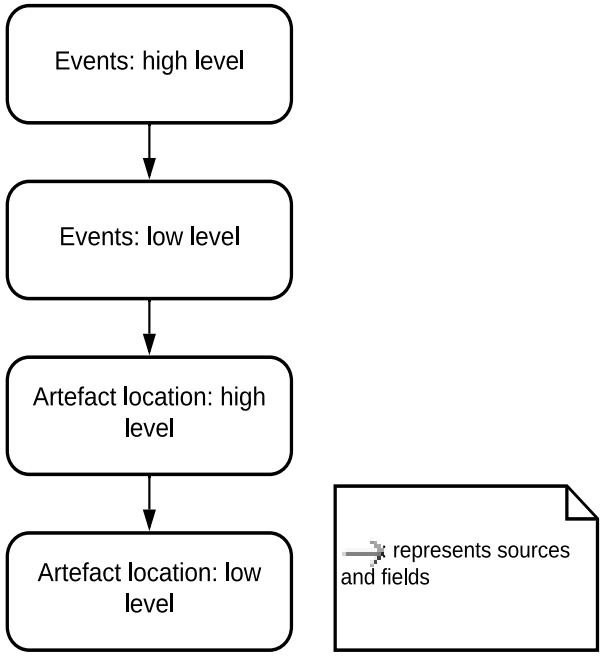


Figure 13. The building block of methodology

ideas behind the breakdown of the timeline into four levels of abstraction are to present the different kinds of information related to different events from the most significant to the least significant, and a different structure should be specified for each level along with distinctive levels of details of information to reduce the complexity of the timeline, omitting unwanted details, enforcing the correctness of timeline and presenting only information that will be helpful to recognise and understand particular actions executed by the users by analysing different sources and fields [16].

The proposed methodology in the research involves the development of four different modules, each module corresponding to one of the levels of abstraction of the timeline of events and artefacts. Each level of the proposed methodology presents a timeline related to different activities or actions performed by a user on a particular digital device. The timeline of the first level, i.e., Events: high level (new entries and web surfing), depicts information related to numerous activities performed on the web and the system locally with a high level of abstraction. It includes browsing web pages, downloading information, composing mail, entries of files created by the user and the system itself. The timeline of the second-level, i.e., Events: low level (web surfing, actions of modifying), illustrate a detailed information related to the web activities and a list of files created by the user and the system. It includes URL addresses of the web pages accessed by the user, mail addresses of the user used to compose and receive mail, size, type of file and many more with a lower level of abstraction. Moreover, the timeline of the third level of methodology, i.e., Artefact location: high level (includes all application files), illustrates the information, which is presented by the first two levels of methodology along with the information related to different applications used by the user and the system itself to perform various types of operations or activities, i.e., locally or online. In the end, the timeline of the fourth level of methodology, i.e., Artefact location: low level, illustrates the entire information, which is presented at the first three levels of methodology with the lowest level of abstraction. In short, each level of methodology is dedicated to presenting relevant information related to different activities performed by the user on a particular computer system along with different level details by examining the distinct sources, fields and parameters. All levels of the proposed methodology are discussed in detail in the sub-sections below.

3.2.1. Events: High Level (New Entries and Web Surfing)

The Events: high level (new entries and web surfing), i.e., the first level of abstraction of the timeline of events and artefacts, presents the information related to events or activities with the highest level of priority in terms of relevance or importance and that can affect the computer system most from an investigation point of view. It includes the events related to different types of files created by the user on a particular computer system, such as docx, txt files, numerous activities performed on the internet or web, such as access of particular web page, access of Gmail or other digital platform accounts, downloads of information in the form of files of various formats (jpeg, pdf and docx) and other vital information with the

highest levels of abstraction of information. This information is provided by six different sources, namely “LNK”, “LOG”, “META”, “OLECF”, “PE” and “WEBHIST”. Six distinct fields, namely “date”, “time”, “source”, “short”, “visit” and “reference”, are considered to reconstruct the timeline at this level. The fields “reference” and “visit” are explicitly added at all (four) levels. The field “reference” stores the reference number to the original source line of the initial file of the timeline. It is used to keep a relationship with the original file. The field “visit” provides a brief depiction of the events and artefacts (has value only for the source WEBHIST).

The source “LOG” is examined to attain vital information, such as MAC address. The source “LNK” is analysed to collect the information related to the files that are created and frequently accessed by the user along with the applications that are most frequently used by a user with a high level of abstraction. The source “META” is considered to attain additional information related to distinct types of files created by the user, such as the address, type and name of the file. Moreover, the information about various applications used by a user to create and access these files is retrieved. For instance, xltx, potx and dotx files are template files used by applications Microsoft Excel, Microsoft PowerPoint and Microsoft Word applications, respectively, to create default layouts, setting such as auto text, toolbars and formatting styles for the new document or file. The source “OLECF” is analysed to collect information related to the system files, data files, plugin files and misc files. Moreover, the system files (.msp) are used to upgrade the Windows operating system and other Microsoft programs. The data files (.dat) are the most common types of computer files. They may be installed with applications or created by the users. Most data files are saved in a binary format, though some store data as plain text. The examples of the data files include libraries, project files and saved documents. The plugin file (.xla) is used to add modules, extra functions and other tools to Microsoft Excel. It may be included with the Excel software program, developed by a third party or created by the user. The misc file (.msi) is a package that contains installation information for a particular installer, such as files to be installed and installation locations. It may be used for Windows updates as well as third-party software installers. The source “PE” is examined to find out the list of the application files (exe, apk and ipa file) that represent various Windows programs used by a user on a particular computer system to perform different kinds of activities. The source “WEBHIST” is analysed to interpret various activities performed by the user on the internet or the web. It includes the identification of different web pages accessed by a user on the internet, information related to the accessed web pages, download of information from the web, access of mail or other accounts. Moreover, it as well contains how (LINK–user clicked a link, GENERATED–selected an entry from the list, RELOAD–user reloaded the page and TYPED–user typed the URL in the URL bar) each specific web page is accessed. Additionally, for the elimination of the repetition of the same event in the timeline, a single event is considered based on the one event per time unit (second) at the Events: high level.

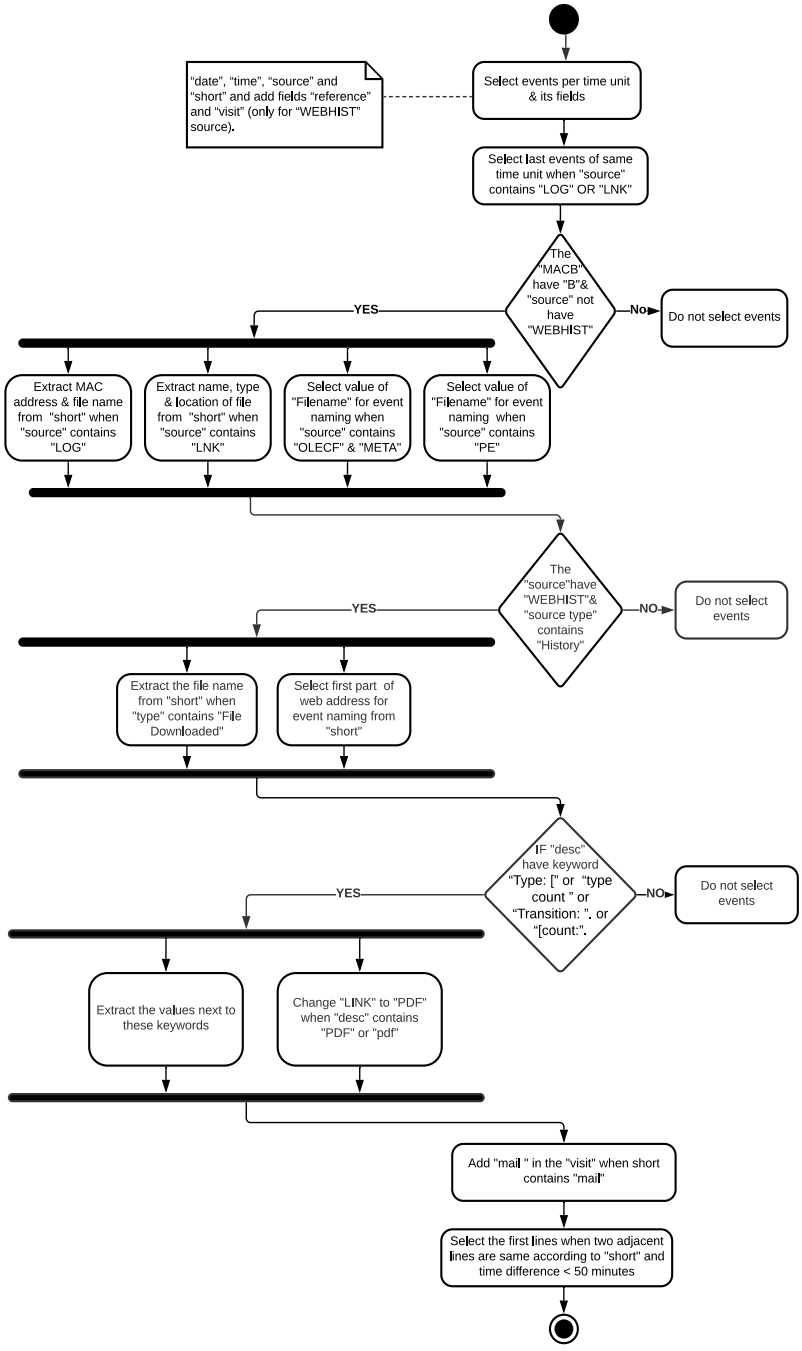


Figure 14. Events: high level (new entries and web surfing)

Figure 14 shows the working of the algorithm of Events: high level (new entries and web surfing). The events are selected according to one event per time unit to remove the duplicity of multiple events at the same time unit, and their six fields, namely “date”, “time”, “source” and “short” and fields “reference” and “visit” (only for “WEBHIST” source), are explicitly added. The value of all (six) fields corresponding to six sources, namely “LNK”, “LOG”, “META”, “OLECF”, “PE” and “WEBHIST”, are extracted from different seventeen fields available in the source file. The similar values for three fields, i.e., “date”, “time” and “source” of all six sources, are extracted and stored from the original source file. The explicitly added field “reference” stores the reference number to the original source line of the initial file of the timeline for all sources. The field “short” stores different values and information corresponding to each source. Only value “B” is considered from the field “MACB” for all sources, except the source “WEBHIST”. It shows entries of new files or documents created (birth) by the user or the system itself.

The MAC address and file name are extracted from the original source file and stored into the field “short” of source “LOG”. The relevant information related to files that are created by a user or a system, such as location, name and type of file, are extracted and stored into the field “short” of source “LNK”. The value of field “filename” is used for event naming and stored into the field “short” of sources “META” and “OLECF”. All application files from the field “filename”, i.e., file end with .exe or .apk or .ipa extension, are extracted and stored into the field “short” of source “PE”.

For source “WEBHIST”, the only value “History” of field “source type” is considered. The information related to the download files can be extracted and stored into the field “short” by analysing the value “File downloaded” from the field “Type”, and then the value of field “visit” is “Download”. The address of the Gmail program can be extracted and stored into the field “short” by searching the keyword “mail” in the field “short”, and the value of field “visit” is “mail”. In other cases, the value of field “short” is the first level of addresses of the web pages accessed by the user, but the value of field “visit” is different and depends upon the web browsers used by the user to access the web pages. In order to extract the values for field “visit”, the field “desc” need to be analysed. If the field “desc” contains keywords “Type: [” or “ type count” or “Transition: ” or “[count:”, then extract the values next to these keywords and store them in the field “visit”. Similarly, if the field “desc” contains the keywords “pdf” or “PDF”, then change the value “LINK” to the value “PDF” and save it in the field “visit”. The possible values of field “visit” are GENERATED, FORM_SUBMIT, TYPED, LINK, RELOAD and many more. In the end, if there is a duplicity of two adjacent lines according to the field “short” and the time difference is less than 50 minutes, then only the first line is considered.

3.2.2. Events: Low Level (Web Surfing, Actions of Modifying)

The Events: low level (web surfing, action of modifying), i.e., the second level of abstraction of the timeline of events and artefacts, provide a timeline with a lower

level of abstraction, more detailed and additional information related to discrete activities performed by the user than the first level of abstraction of the timeline, i.e., Events: high level. The timeline of this level illustrates more detailed and relevant information related to the activities performed locally or online. It includes attaining the complete URL address of the web pages, mail addresses of users used for communication, information related to different files and applications are created and accessed by the user and system itself, such as author, size, type of file and much more. This information is extracted and illustrated by analysing nine different sources and considering seven distinct fields at the Events: low level (web surfing, action of modifying). The three other sources are added, namely "FILE", "REG" and "RECBIN", and one more field "extra" along with the same sources and fields used at the Events: high level. The six sources, namely "LNK", "LOG", "META", "OLECF", "PE" and "WEBHIST", are considered at both Events: high level and Events: low level. The timeline at Events: low level provides relatively the same information as provided by the timeline at Events: high level but with the addition of more detailed and vital information. At Events: low level, the source "META" provides more detailed information related to different files that are created and accessed by the user along with the applications that are used by a user to create and access these files. It includes authors of files, names of applications used to create different files, versions of used applications and many more. The source "OLECF" provides a more detailed information related to the system files (.msp), data files (.dat), plugin file (.xla) and misc file (.msi), such as name, author and version of these files used to upgrade Windows operating system and Windows programs, add functionalities or plugins in the Windows programs. The source "WEBHIST" provides more precise and detailed information related to the actions performed by a user on the web along with the information provided at Events: high level. It includes the title of content search by the user on the internet, such as "http://www.htmlpublish.com/convert-pdf-to-html", the more specific address of a particular web page accessed by a user and the usage of Gmail by a user. The three sources: "LNK", "LOG" and "PE", provide the same information at both levels of reconstructed timeline, i.e., Events: high level and Events: low level.

The source "FILE" provides information related to the files that are accessed, created and modified by the operating system. It includes location, size and type of file. Moreover, it as well provides information related to browsing activities performed by using a browser. The source "REG" provides information about the data files (.dat). A .dat file s a generic data file created by a specific application. It may contain data in binary or text format (text-based DAT files can be viewed in a text editor). The .dat files are typically accessed only by the application that created them. At last the source "RECBIN" provides information about the files that are deleted by the user from the system and recycle bin. Similar to the Events: high level for the elimination of the repetition of the same event in the timeline, single event is considered based on the one event per time unit.

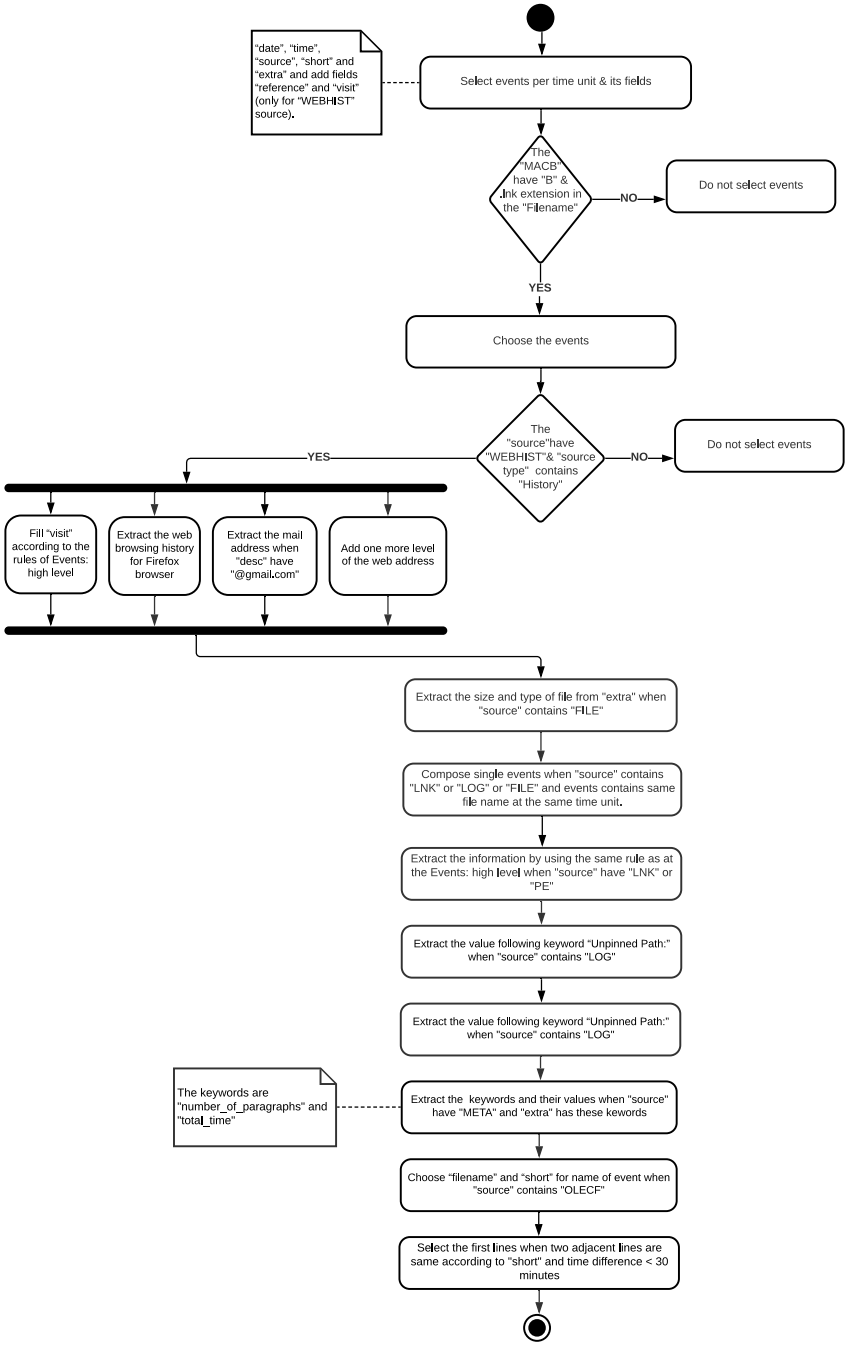


Figure 15. Events: low level (web surfing, actions of modifying)

Figure 15 shows the working of the algorithm of Events: low level (web surfing, action of modifying). The events and their fields are selected according to one event per time unit. At this level, seven fields with the addition of one more field “extra” and nine sources with the addition of three sources, namely “FILE”, “REG” and “RECBIN”, along with the same sources and fields used at the Events: high level, are considered. The value “B” of the field “MACB” and value ends with “.lnk” extension are considered to include a line of events. Moreover, for the source “REG” and “WEBHIST”, select the first event, if multiple events are repeated at the same time unit. Similarly, select the last event of source “LOG”, if multiple events are repeated at the same time unit, corresponding to “REG”, “OLECF” and “LOG” sources. The field “visit” contains value only for source “WEBHIST”, and field “extra” contains value only for sources “FILE”, “META”, “OLECF”, “WEBHIST” and “RECBIN”.

The value of the four fields, namely “date”, “time”, “source” and “reference”, for all nine sources are extracted and stored similarly as at the Events: high level. The value of fields “visit” and “short” of source “WEBHIST” are stored as well according to the same rule as at the Events: high level. For the value of field “extra” of source “WEBHIST”, the field “extra” of the source file is analysed. If the field “source type” contains the value “Firefox History”, then search the field “extra” for keyword “username=”, then extract this keyword and its value. In other cases, extract the value till the keyword “schema_match”. The mail address of the user can be extracted and filled into the field “extra” by searching the keyword “@gmail” in the field “desc”, and one more level of the address of the web page is added and filled into the field “extra” in case of value of field “source type” is “Chrome History”.

For the source “FILE”, the value of the field “short” from the source file is used for the name of the event and is stored into the “short” field. In order to fill the field “extra”, the field “extra” need to analyse and store value till keyword “sha256_hash”. The single event is composed in case of repetition of multiple events, at the same time, unit corresponds to the sources “LNK”, “LOG”, “FILE”, and the events shows the same file name. The value of the field “short” of sources “LNK” and “PE” are extracted and stored similarly as at the Events: high level. For the source “LOG”, the value of field “short” is stored by analysing the field “short” and searching for the keyword “Unpinned Path:”. If the keyword is available, then extract the value following the keyword. In other cases, just extract the entire value from the field “short”.

For the source “META”, the field “filename” is used for the name of the event, and its value is stored in the field “short”. The fields “extra” and “short” are analysed and searched for keywords “number_of_paragraphs”, “total_time” and “Creating App:”. If these keywords are available, then extract these keywords along with their value and store them in the field “extra”. In other cases, store the value of field “Short” in the field “extra”. For the source “REG”, only the value “Registry Key: UserAssist” of the field “source type” is considered and choose one line per time unit. The value of field “short” is stored and used for the name of the event. For

source “OLECF”, the value of the fields “filename” and “short” are stored in fields “short” and “extra” and used for the event naming. The single line per time unit is selected for source “RECBIN”. In the end, if there is the duplicity of two adjacent lines according to field “short” and the time difference is less than 30 minutes, then only the first line is considered.

3.2.3. Artefact Location: High Level (Include All Application Files)

The Artefacts location: high level (includes all application files), i.e., the third level of abstraction of the timeline of events and artefacts, provides a timeline with a lower level of abstraction and more detailed information related to various activities performed by a user on a computer system. The timeline of this level illustrates detailed information related to all types of operations performed by the user along with all types of applications and their details used by a user and the system to perform different operations, such as modified, access, change, creation (birth), updating and many more. This information is extracted and illustrated by analysing nine different sources and considering ten distinct fields by adding three more fields, namely “MACB”, “source type” and “desc”, to reconstruct the timeline at Artefact: high level (include all application files).

The timeline at this level provides additional detailed information by including the above mentioned three more fields. The field “source type” provides a description of sources, field “MACB” present a particular file or document that is either modified, accessed, changed or created (birth) by the user or the system itself, and the field “desc” provides complete descriptive information of the activities performed on the system by a user.

For instance, for the source “WEBHIST”, the value of the field “source type” can be either “Firefox History” or “Chrome History”. This shows that the user uses either Mozilla Firefox or “Google Chrome” web browser to perform activities on the internet or web, such as access to information or download information from the internet. Similarly, for each source, the field “source type” contains the vital value to provide descriptive information related to the actions performed by the user. Moreover, this level as well assists the digital investigator to identify various kinds of applications used by the user to perform operations and files used by the operating system itself. In short, the Artefact location: low level provides a partially complete and descriptive overview of various activities performed by the user, i.e., online and offline activities, using applications and the files used by the system itself.

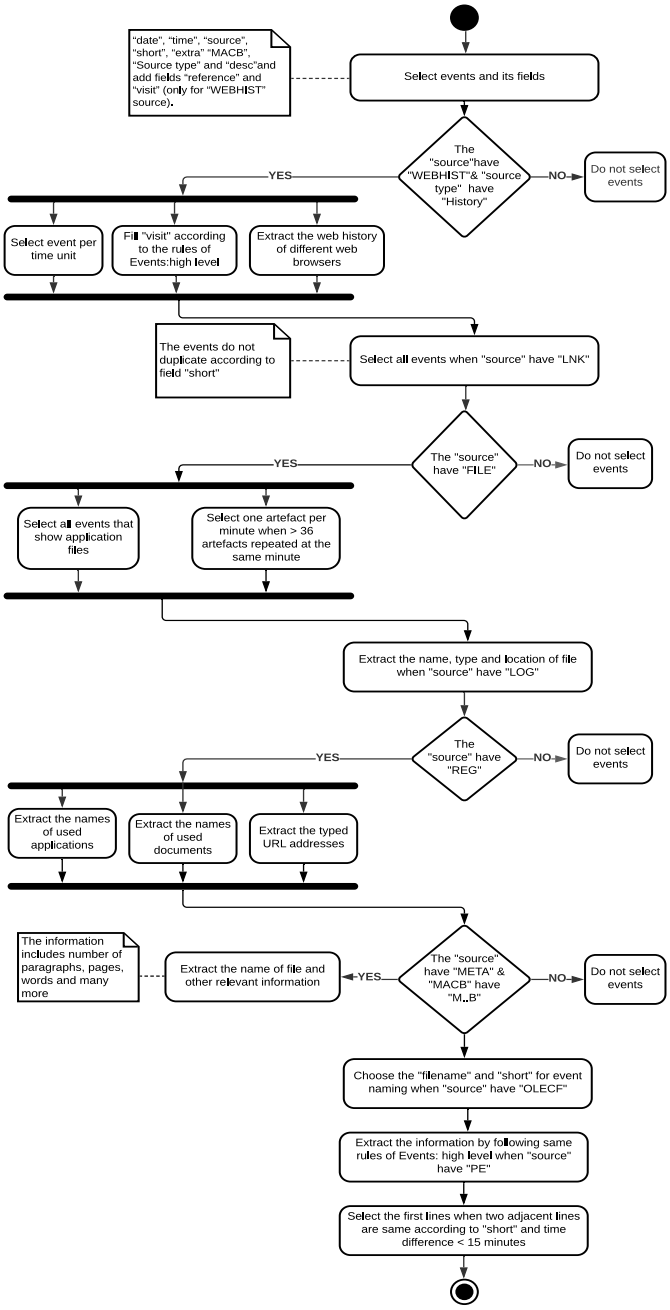


Figure 16. Artefact location: high level (include all application files)

Figure 16 shows the working of the algorithm of Artefact location: high level (includes all application files). Select events and their different ten fields with the addition of three more fields, namely “MACB”, “source type” and “desc” along with the same seven fields used at the Events: low level. The number of the sources will remain the same as at the Events: low level. The value of four fields, i.e., “date”, “time”, “source” and “reference” of all sources, are extracted and stored similarly as at the first two levels. The similar values from the input files are extracted and stored into two new added fields, i.e., “MACB” and “source type” of all sources. The value of fields “visit” and “extra” of source “WEBHIST” are extracted and stored similarly as at the Events: low level. The similar values are stored in the fields “short” and “desc” of source “WEBHIST”, which are extracted from the input file. Moreover, only a single event per time unit is considered. For source “LNK”, it includes all lines but does not include the duplicity of lines according to the field “short”. The same value from the input file is extracted and stored in the field “desc”, and the values for the other fields are extracted similarly as at the Events: low level.

For the source “FILE”, consider all lines which show executable files but do not include the duplicity of artefact when there are more than 36 artefacts continuously repeated at the same time unit. In other cases, include single artefact per time unit. The value for all fields are stored and extracted in the same manner as at the Events: low level expects field “desc”. For this, the same value is extracted and stored in this “desc” field. For source “LOG”, include one artefact per time unit when field “short” does not begin with the keyword “Entry” and include all artefacts whose field “format” contains value “lnk”. The value for all fields are stored and extracted in the same method as at Events: low level expects field “desc”. The name of the file or the value of the field “filename” is stored in the field “desc”. Moreover, do not include artefact of source “LOG” when the artefacts are repeated, corresponding to the source “LOG” and “REG” at the same time unit. Moreover, do not include the duplicity of an artefact of the source “LOG”, according to the field “short”, i.e., the name of the file.

For the source “REG”, all artefacts are included, and different type of information is extracted from the field “desc”, corresponding to different values in the field “source type”. The name of the used applications can be extracted from the field “desc” when the value of the field “source type” is “Registry Key: UserAssist” or “UNKNOWN : MRU List”. Similarly, the information related to the used documents and URL addresses can be extracted when the value of field “source type” is “UNKNOWN : MRUListEx” or “UNKNOWN : Typed URLs” or “UNKNOWN”. The value for all fields are stored and extracted in a similar method as at Events: low level expects the fields “desc”, “short” and “extra”. The value of the field “filename” is stored in the field “short”. A similar value is stored in the field “extra” from the input file. The field “desc” contains different information related to the documents, applications and URL address.

For the source “META”, use the field “filename”, as it shows the real file name. Extract the information related to the file, such as the author of the file from

the field “short”, keywords “number of paragraphs”, “total time” and their values from the field “extra” and keyword “Number of pages”, “Number of words”, “Number of characters”, “number of lines” and their values from the field “desc”. The value of all fields are stored and extracted in a similar method as at the Events: low level expects the field “desc”. The value for field “desc” is the above-mentioned keywords and their values. For the source “OLECF”, select the last line of the same time when multiple lines are repeated at the same time. The values for all fields are stored and extracted in a similar method as at the Events: low level expects the fields “desc”, “short” and “extra”. For the value of fields “short” and “extra”, the values of fields “filename” and “short” are stored. At the end of the field “desc”, the first 80 characters are stored. For the source “PE” and the value for all its fields are stored, according to the same method as at the Events: low level. In the end, if there is the duplicity of two adjacent lines, according to the field “short”, and the time difference is less than 15 minutes, then only the first line is considered.

At the Artefacts location: low level, i.e., the fourth and the last level of abstraction of the timeline of events, provide a timeline with the lowest level of abstraction and complete detailed information related to different activities performed by a user on a particular computer system. At this level, all nine sources are analysed, and all eighteen fields are considered with the addition of nine fields, namely “timezone”, “type”, “user”, “host”, “version” “filename”, “inode”, “notes” and “format”. Each level of timeline reconstructed by the proposed approach provides a timeline with different levels of abstraction and detail of distinct information related to different activities performed on a system by a user.

3.3. Novel Ontology Based on the Proposed Methodology

Digital forensic tools generate unstructured timelines from various sources of data. The unstructured timelines are difficult to interpret because of cognitive overload and the diversity of semantics. Thus, digital practitioners are not able to understand the newly encountered terminologies and attain evidence. In order to figure out these issues, a promising approach containing correct and reliable representations that allow the user to structure data and standardize their representation is required. For this, a digital forensic approach backed by a knowledge model called the ontological approach was developed. This approach facilitates the correct representation of a digital incident and other actions that are taken during the investigation to get the results. A structured and formal knowledge presentation allows the formation of automatic processes more conveniently by composing information in a way that is understandable by the machine and provides an easy way for digital investigators to query, interpret and visualize the information [17].

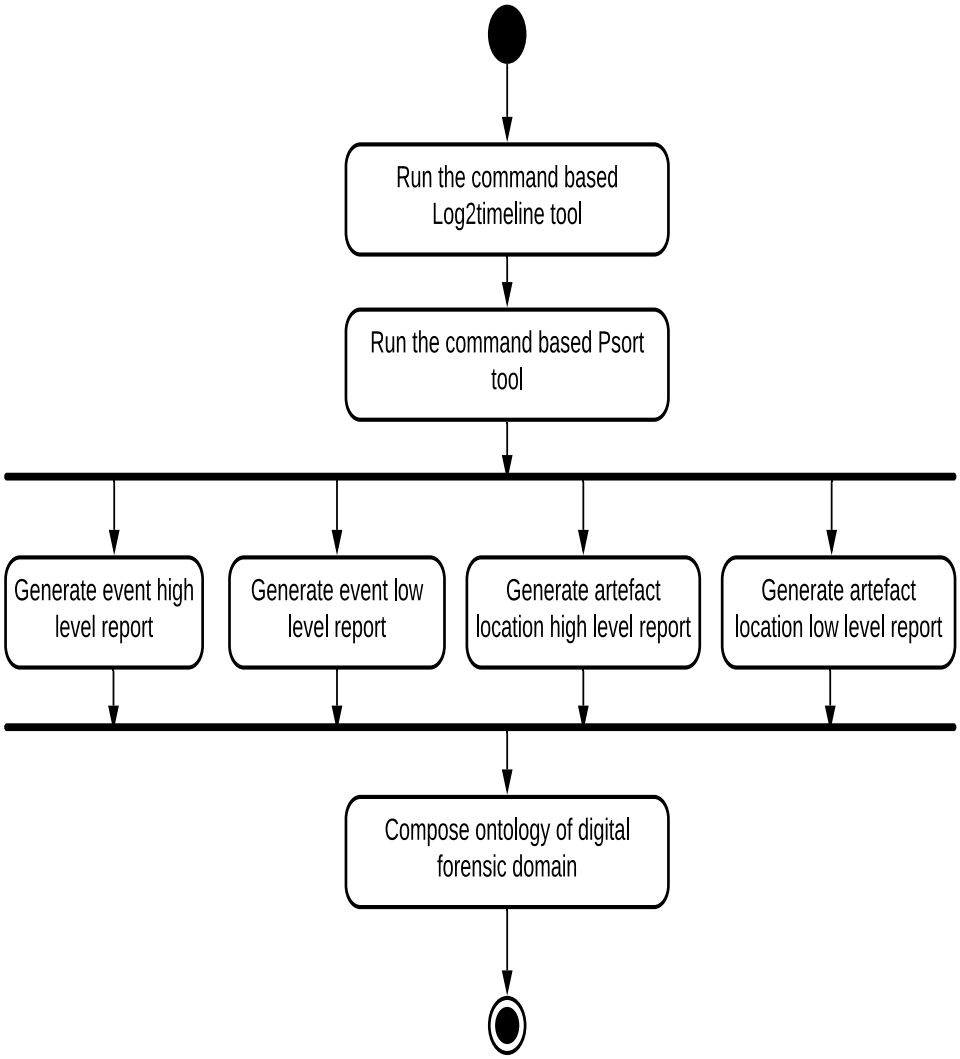


Figure 17. Timeline reconstruction with ontology composition

An ontology is a prototype of the representation of knowledge of domains by structuring this knowledge, using classes or entities, relationships and constraints. The other features of ontology are that it allows automatic reasoning of data by showing a vital relationship among concepts. Secondly, the ontology supports very coherent and easy navigation that the user moves, using available concepts in the ontology [17]. Thirdly, the ontology can represent any form of data, namely structured and unstructured data. The developed ontology was as well supported by the novel abstraction based approach, as shown in Figure 17. All these characteristics of the ontological approach are combined to correctly represent the knowledge generated during the investigations, assist the practitioner in interpreting

the information, attain evidence from the timeline and identify the causes of digital accidents [17].

The main ideas behind the development of a novel ontology for digital forensics are first to assist practitioners in understanding the new terminologies and connections among them that are encountered during the investigation. A second idea behind the ontology development is to share the domain (digital forensics) knowledge among the researchers, digital practitioners and users. The author has reviewed the timeline and encountered new terms, their properties and relationships among them. The abstraction based approach was implemented on Windows, Android and iOS-based operating systems devices to identify various new terms that the developed ontology would depict the maximum number of terms and relationships, which are identified on typical Windows, Android and iOS operating systems based devices [17].

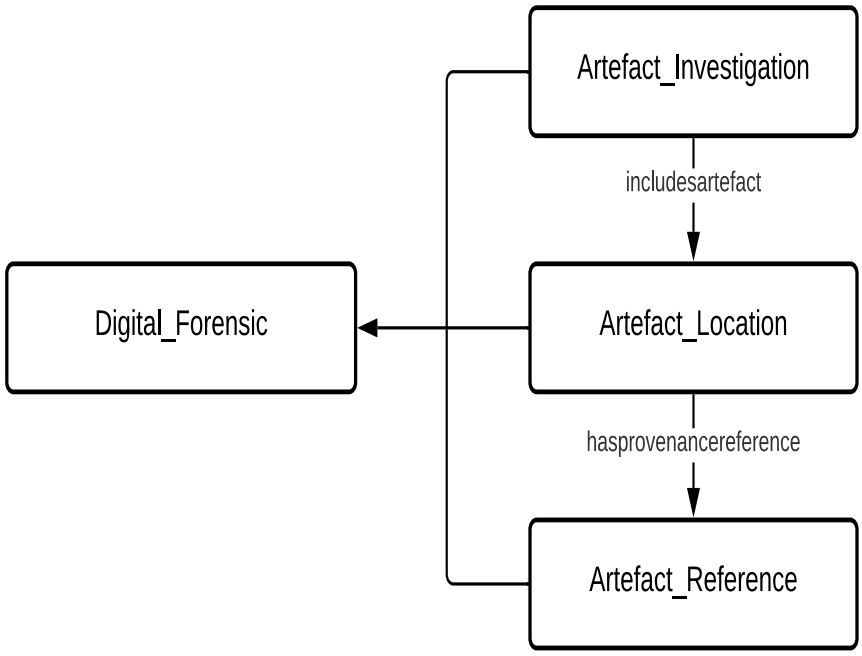


Figure 18. Classes and object properties [17]

In practical terms, developing a particular ontology consists of the following major steps:

1. To define the classes,
2. To organize classes in a taxonomic hierarchy,
3. To describe slots and values,
4. To fill the values into slots.

Lastly, an individual instance of the classes was defined, the values were filled in the slot, and the restrictions were defined to create a knowledge base. In

the developed ontology, the top-down roach was used to define the classes and organize the taxonomy of classes. It begins with the explanation of the most general concepts in the domain, followed by an explanation of specialization concepts [17].

Digital _Forensic is a base class that represents the general concept in the forensics domain, for which ontology is being developed to define common vocabulary, share information and reuse and analyse domain knowledge. Three subclasses are defined from the base class to represent more specific concepts of the digital forensics domain: Artefact_Investigation, Artefact_Location and Artefact_Reference. In order to show the relationship between these classes, two object properties are defined, such as includesartefact and hasprovenancereference, as shown in Figure 18, and to show the relationship between an instance or an individual and a data value; 18 data properties were defined as shown in Figure 19[17].

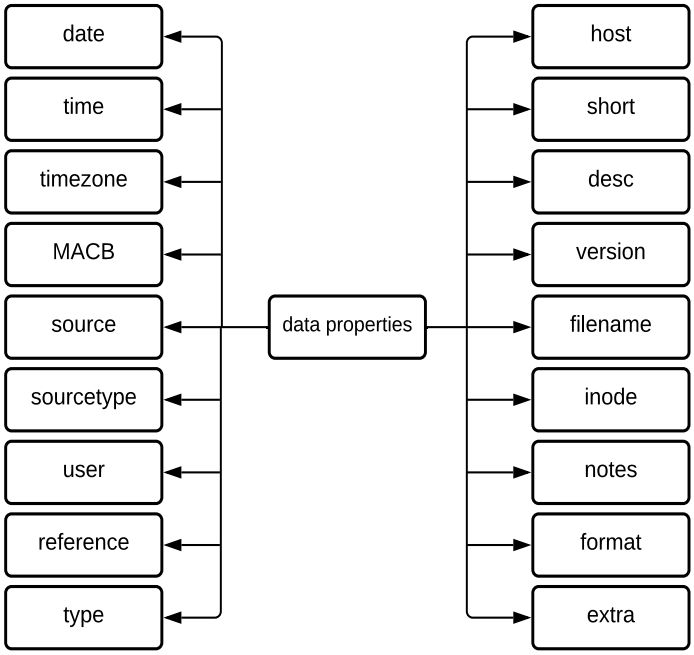


Figure 19. Data properties [17]

3.3.1. Artefact Investigation

The first sub-class Artefact_Investigation defines the basic terms related to the digital forensics process, and different actions are to be taken at various stages by different specialists during the investigation of the case. In ontology, each concept is described by defining a class. Thus, in order to define the sub-terms related to the Artefact_Investigation subclass, new subclasses are defined as shown in Figure 20. A digital attack is committed against an individual or group of individuals intentionally with a criminal motive to harm physically or mentally is known as a victim. The subject provides descriptive information about a specific attack that is committed and forensic processes, such as details of the victim, investigator, examiner, digital attack, and many more [17].

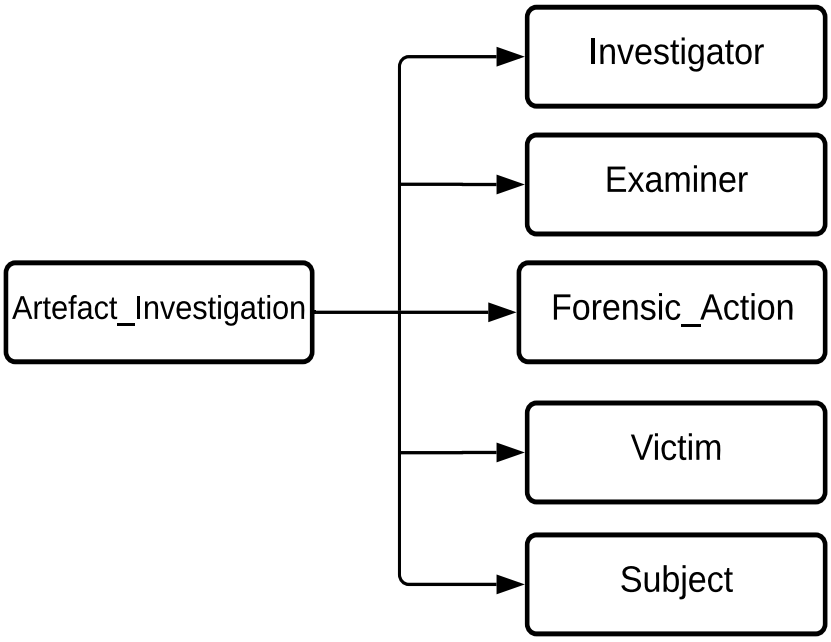


Figure 20. Artefact investigation [17]

Forensic Action represents a recognized scientific and forensics process used by the digital practitioners in digital forensics investigations to collect evidence from the digital devices. It is a multiphase process beginning from the recognition of digital devices as potential digital evidence to the stage where it is demonstrated as evidence in a court of law. The sequence of various phases of the digital forensics process is shown in Figure 21 [15]. The forensics investigator is the person who is initially responsible for examining the captured evidence from the scene of the incident. The investigator documents several types of captured data and provides research on the parameters and technical specifications of the data storage devices

and details of various data components of the evidence, as it is presented for him/her by the data capture specialist. Then, the forensics investigator provides his/her report in the form of evidence to the forensics examiner who is independently responsible for analysing this evidence. The forensic examiner job is to provide logical conclusions about the dataset and what it reveals as to the nature and purpose of the evidence [17].

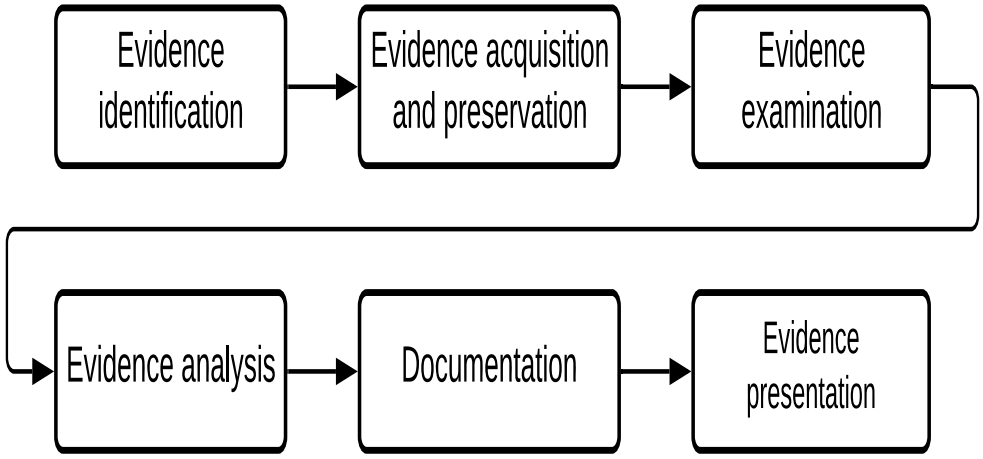


Figure 21. Forensics action [16]

3.3.2. Artefact_Location

The Artefact Location subclass represents a novel abstraction based approach for the analysis of the timeline. The abstraction based approach consists of four abstraction levels of the timeline, as shown in Figure 22, namely Events: high level, Events: low level, Artefact location: high level and Artefact location: low level. At each level of abstraction, different number of sources and fields are considered to analyse the timeline. The number of sources to be considered at each level can be varied and is dependent on the maximum number of sources that are available in the timeline. At Events: high level, the information related to the creation of new files and web surfing activities, such as access of web pages and download files, is provided with a higher level of abstraction, considering six different fields, which compose a unique structure [17].

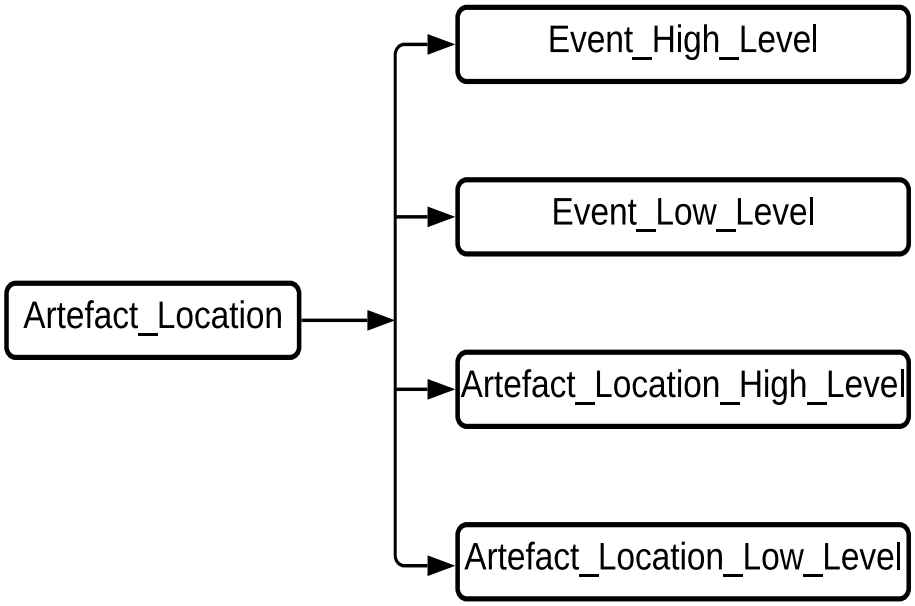


Figure 22. Abstraction approach [17]

At Events: low level, detailed information is provided related to the distinct categories of activities that are performed by the user related to the file, such as modification, access, changes and birth and web surfing activities. It as well includes the addresses of particular web pages that are accessed by the users and the downloaded information from the web. At this level, seven different fields are considered to compose and analyse a timeline. The first two levels, i.e., Events: high level and Events: low level, provide information related only to the user activities. In order to have a clear view of the timeline and activities, additional information is required, such as a list of all executable files executed by a user, applications and files that are regularly accessed by the user and the authors of files. Besides, detailed information is as well provided, related to distinct activities performed by the user on the web. For example, in the case of web browsing activity, detailed information will include the name of the used web browser and how (LNK – user follows a link, TYPED – user type the URL, RELOAD – user refresh the web page) the web page is accessed. Such detailed and relevant information is provided at the Artefact location: high level and Artefact location: low level by analysing ten and seventeen fields, respectively [17].

3.3.3. Artefact_Reference

The Artefact_Reference subclass shows different types of sources of data that are encountered during the analysis of the timeline. The number of sources varies, depending on the operating system and its version and quantity of data. However, the source will support the investigator in recognizing information available in the timeline and the different actions performed by the user on a particular device efficiently [17].

The novel ontology for the digital forensic domain is based on the Log2timeline, and Psort tool is composed of three major subclasses: Artefact_Investigation, Artefact_Location and Artefact_Reference. The subclass Artefact_Investigation presents a recognized scientific and forensics process used by different specialists during the investigation of the case to interpret the digital evidence. The second subclass Artefact_Location shows a novel abstraction based approach for the analysis of the timeline. In the end, the third subclass Artefact_ Reference shows the newly encountered terminologies during the analysis of the timeline, and these terms are discussed in detail in chapter 4[17].

3.4. Summary

It can be summarized that there are numerous issues associated with the reconstruction and analysis of the timeline of events in order to interpret various activities, i.e., web or online activities and offline activities performed by the user on a typical digital device, it includes innovation in technology frequently, a heterogeneous, huge quantity of data and manually-performed event reconstruction process, which is inefficient and expensive. Moreover, the available approaches are as well not able to fulfil and address the five major issues in the digital forensic investigation, which are highlighted in chapter 2. Thus, a novel approach for the analysis of timeline, based on the timeline generated by two digital forensics tools, namely Log2timeline and Psort, is proposed.

The proposed abstraction based approach splits the timeline into four levels of events and artefacts, namely Events: high level (new entries and web surfing), Events: low level (web surfing, actions of modifying), Artefact location: high level (includes all application files) and Artefact location: low level. The main idea behind the breakdown of the timeline into four levels of abstraction is to present different kinds of information, and a different structure should be specified for each level along with distinctive levels of details of information to reduce the complexity of the timeline, omitting unwanted details, enforcing the correctness of the timeline and presenting only information that will be helpful to recognise and understand particular actions executed by the users by analysing different sources and fields. The capability of the proposed approach for the analysis of the timeline is demonstrated by implementing it on different operating systems based devices and comparative studies among the proposed and existing approaches, and it is discussed in the following chapter 4.

Numerous features of the ontology compose information in a way that is understandable by the machine and provides an easy way for the digital investigators to query, interpret and visualize the information. Some of the features are structured and formal knowledge presentation, representation of knowledge of domains by structuring this knowledge, using classes or entities, relationships and constraints, automatic reasoning of data and many more. The main ideas behind the development of a novel ontology for the digital forensics domain are, first, to assist the practitioners in understanding new terminologies and connections among them that are encountered during the investigation of a digital crime; secondly, to share the domain (digital forensics) knowledge among the researchers, digital practitioners and users.

The developed ontology contains one parent class, i.e., “Digital _Forensic”, which represents the digital forensics domain, for which the ontology is being developed, to define common vocabulary, share information and reuse and analyse domain knowledge. From the base class, three subclasses are defined to represent more specific concepts of the digital forensics domain: Artefact_Investigation, Artefact_Location and Artefact_Reference. In order to show the relationship between these classes, two object properties are defined, such as includesartefact and hasprovenancereference, to show the relationship between an instance or an individual and the data value, 18 data properties. The first sub-class Artefact_Investigation defines the basic terms related to the digital forensics process, and different actions are to be taken at various stages by different specialists during the investigation of the case. The Artefact_Investigation sub-class contains five child classes, namely victim, investigator, examiner, subject and Forensic_Action. The second class Artefact_Location represents a novel abstraction based approach for the reconstruction of the timeline. This subclass contains four child classes, namely Event_High_Level, Event_Low_Level, Artefact_Location_High_Level and Artefact_Location_Low_Level. The last and third sub-class Artefact_Reference represents the new and different terminologies that are encountered during the analysis of the timeline using an abstraction based approach. The different terminologies corresponding to the different operating systems based devices are shown in the proposed ontology and are discussed in chapter 4.

4. EXPERIMENTAL STUDIES

4.1. Overview

This section is devoted to illustrating the findings of the abstraction based approach for the timeline analysis, novel ontology and visualization technique. Moreover, this section contains three major sections and various sub-sections.

- The "Finding of abstraction based timeline analysis approach" section presents the results of the abstraction approach, corresponding to Windows, Android and iOS operating systems. This section consists of several sub-sections.
- The "Finding of novel ontology" section illustrates the outcome of novel ontology along with the evaluation of ontology. Moreover, this section contains six sub-sections.
- The "Finding of visualization" section is devoted to showing the outcome of visualization techniques corresponding to Windows, Android and iOS operating systems. This section comprises five sub-sections.

4.2. Finding of Abstraction Based Timeline Analysis Approach

4.2.1. Experimental Setup

In order to illustrate the capabilities of a novel abstraction based approach for the analysis of the timeline, several experiments were performed and implemented in object-oriented programming language i.e., Java. For this, the abstraction based approach was implemented on different operating systems [5]based devices, namely Windows, Android and iOS operating systems. Table 7 shows additional information corresponding to these operating systems.

Table 7. Configuration of operating systems

Operating system	Version	File system	Size of data
Windows	Windows 10, 1909	NTFS (new technology file system)	150 GB (Approx.)

[5] *Some passages have been quoted verbatim from the following source:*

An ontology based on the timeline of Log2timeline and Psort using abstraction approach in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

Android	Android, L5.1.1	EXT4 (extended file system)	12 GB (Approx.)
	Android, 9 PKQ1.180904.001	EXT4 (extended file system)	20 GB (Approx.)
iOS	iOS 13.3	APFS (apple file system)	20 GB (Approx.)

Initially, distinct types of operations are performed on the above mentioned operating systems based devices, which are categorized into two main categories as described below (see Table 9 and Table 8) to generate timeline by using command-based digital forensics tools, i.e., Log2timeline and Psort, then the generated timeline is analysed by implementing the proposed approach.

4.2.1.1. Online Operations

It includes all types of actions performed on the internet, such as access to the web pages, downloads of information and composing an email by using any browser as shown in Table 8.[6]

Table 8. Scenario summary (web history case study) [17]

Experiment	Scenario Summary
1.	Flat or Blank execution of Internet Explorer, Google Chrome and Mozilla Firefox, i.e., running without opening/surfing any webpage by typing 'about blank' in the URL address text box. The experiment was conducted by launching all three browsers and then instantly closing their windows. The purpose of the flat execution of IE was to check which of the system and temporary files are accessed by the browsers when no webpage is loaded**.**

[6] *Some passages have been quoted verbatim from the following source:*

An abstraction based approach for reconstruction of timeline in digital forensics
Bhandari, S. & V. Jusas.
Symmetry, 2020

2.	Launched all three browsers parallelly and sequentially and opened Kaunas University of Technology website 'https://ktu.edu'. Checked personal email and then signed out, followed by closing the respected browsers window.
3.	Launched browser and opened various sessions for random surfing of academic, commercial, social and news websites like "www.linkedin.com", "www.forensicswiki.org", "www.sciencedirect.com", "www.elsevier.com", "https://www.seb.lt/", "www.researchgate.net" and many more. Further on, performed some activities, such as downloading required files and entering comments to some portals and closed browser after a session of different times.
4.	Launched the browser and opened "https://mail.google.com/" websites to check the emails. At this time, the emails were checked, and the attachments were downloaded and uploaded, then subsequently opened through their associated application programs like MS-Word, Adobe Acrobat Reader, Excel and PowerPoint.

4.2.1.2. Offline Operations

It includes all types of actions performed on a typical computer system, such as creation, modification, access, rename and copy of the file by using various applications, such as word pad, Notepad and many others, as shown in Table 9. The experiments were conducted by running each application sequentially and together parallelly.

Table 9. Scenario summary (execution of multiple application programs case study) [17]

Experiment	Scenario Summary
1.	The following application programs were independently loaded, without opening a single file in these applications, in the following order: Internet Explorer, Mozilla Firefox, Google Chrome, WordPad, Notepad, MS-Word, Excel.
2.	The above-mentioned applications were individually run, and only a single file was opened for each application followed by closing the application program.

3.	The above-mentioned group of application programs was discretely executed, and multiple data files were accessed for each individual application. This time, the attempts were made to open the files having an incompatible format to the corresponding application to observe the file system activity patterns of application programs for failed attempts during the loading of inconsistent files.
4.	Multiple applications are executed parallelly, and multiple files were opened for each application. The timings of launching these applications were noted down that the file system activity data sets could be linked to the corresponding application programs.
5.	Executed some operations, such as modified, accessed, Changed, Birth (file creation time) rename, copy and delete, for the above mention applications, namely WordPad, Notepad, MS-Word, Excel.
6.	Insert USB and create a directory in USB and copy some text files from the computer system to the created directory and vice versa. Perform some operations, such as modified, accessed, Changed and Birth (file creation time).

4.2.2. Results

The two case studies of each operating systems based timeline, namely Windows, Android and iOS, are presented to illustrate that the proposed approach is capable of analysing the timeline generated by Log2timleline and Psort digital forensic tools. The proposed abstraction based approach for the analysis of timeline splits the generated timeline into four levels of the timeline of events and artefacts to compose a relevant and recognizable timeline. Thus, the final outcome of the proposed approach is as well in the form of four levels of timeline of events and artefacts. At each level, different numbers of events and fields are considered, and numerous mechanisms are enforced by analysing the timeline to compose a user or digital practitioner understandable timeline and easy to attain digital evidence from the timeline. The generated timeline, corresponding to these operating systems, includes numerous issues, such as the immense volume of data, the duplicity of a large number of events and artefacts at the same time unit, irrelevant and unorganized information and many more. All these issues compose a complex timeline, and digital investigators are not able to understand the timeline and find digital evidence.

Thus, a digital practitioner is lost in a huge number of events and artefacts, unwanted (noise) information and unorganized information in the timeline.

4.2.2.1. Windows Operating System

The two case studies of the Windows operating system i.e., sources “WEBHIST” and “LNK”, are considered. The first case study, namely source “WEBHIST”, is analysed to provide relevant and organized information related to the web activity performed by a user on a particular digital device. Figures 23 and 24 show the outcome of the proposed approach in the form of four levels of timeline of events and artefacts, corresponding to the source “WEBHIST”. At each level of approach, different number of fields is considered; different mechanisms are stipulated to compose an easily understandable timeline. In fact, 6, 7, 10 and 18 different fields are considered at Events: high level, Events: low level, Artefact location: high level and Artefacts location: low level, respectively, along with the implementation of distinct mechanisms at each level to address other issues in the timeline. The first two levels of approach provide an overview of a specific activity performed by the user; it contains the address of the web page browsed by a user and the type of web page visited by a user.

Events: high level (level 1)
Date: 11/14/2017
Time: 11:46:15
Source: WEBHIST
Short: https://mail.google.com
Visit: mail
Reference: 257192

Events: low level (level 2)
Date: 11/14/2017
Time: 11:46:15
Source: WEBHIST
Short: https://mail.google.com
Visit: mail
Extra: https://mail.google.com/mail
Reference: 257192

Artefact location: high level (level 3)
Date: 11/14/2017
Time: 11:46:15
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Short: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 201..
Visit: mail
Extra: https://mail.google.com/mail
Desc: sanxxxxxx525@gmail.com
Reference: 257192

Artefact location: low level (level 4)
Date: 11/14/2017
Time: 11:46:15
Timezone: UTC
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Type: Last Visited Time
User & Host: -
Short: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 201...
Desc: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 2017 For 380 Vacancies Apply Now - sanxxxxxx525@gmail.com - Gmail) [count: 0]
Host: mail.google.com Type: [LINK - User clicked a link] (URL not typed directly - no typed count)
Version: 2
Filename: OS:C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\History
Inode & Notes: -
Format: sqlite/chrome_history
Extra: schema_match: False; sha256_hash: 582bcc588c7bc39ce0d789951fde1bd8296982a04d8a26eb09a582a901302ae3
Reference: 257192

Figure 23. The output of WEBHSIT source of Windows operating system [17]

Keys:
URL: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 2017 For 380 Vacancies Apply Now - sanxxxxxx525@gmail.com - Gmail)
Search Term: mail
Browser: Google Chrome
Description: LINK - User clicked a link

Figure 24. The output of WEBHSIT source of Windows operating system [17]

Events: high level (level 1)
Date: 11/07/2017
Time: 20:11:35
Source: LNK
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Reference: 223117

Events: low level (level 2)
Date: 11/07/2017
Time: 20:11:35
Source: LNK
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Extra: -
Reference: 223117

Artefact location: high level (level 3)
Date: 11/07/2017
Time: 20:11:35
MACB: ...B
Source: LNK
Source type: Windows Shortcut
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Extra: -
Desc: Empty description] File size: 57 File attribute flags: 0x00000020 Drive type: 3 Drive serial number: 0xc04f69f2 Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt
Reference: 223117

Artefact location: low level (level 4)
Date: 11/07/2017
Time: 20:11:35
Timezone: UTC
MACB: ...B
Source: LNK
Source type: Windows Shortcut
Type: Creation Time
User & Host: -
Short: [Empty description] D:\Doctorate Studies\chrome history window 7.txt
Desc: [Empty description] File size: 57 File attribute flags: 0x00000020 Drive type: 3 Drive serial number: 0xc04f69f2 Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt
Version: 2
Filename: OS:C:\Users\User\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\5f7b5f1e01b83767.automaticDestinations-ms
Inode & Notes: -
Format: olecf/olecf_automatic_destinations/lnk
Extra: birth_droid_file_identifier: 6df44ae9-c4d4-11e7-8ac6-a0afbdac1ec0; birth_droid_volume_identifier: a6ab9a4e-a31c-4e48-9416-b0cb2766758a; droid_file_identifier: 6df44ae9-c4d4-11e7-8ac6-a0afbdac1ec0; droid_volume_identifier: a6ab9a4e-a31c-4e48-9416-b0cb2766758a; sha256_hash: a39a0b9e3a0344d2feddf8168148344eb466887e864e8fcc0a276c559b3d11a7
Reference: 223117

Figure 25. The output of LNK source of Windows operating system [17]

Keys:
Address and Name:
D:\Doctorate Studies\chrome history window 7.txt
Source type: Windows Shortcut
Description: Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt

Figure 26. The output of LNK source of Windows operating system [17]

The last two levels of approach provided more detailed and vital information to understand the particular activity more precisely. It includes the complete web address of the web page accessed by the user, the name of the web browser used to access the web page and how a particular web page is accessed. One can easily understand the timeline related to the web activity performed by a user and analyse the information to obtain digital evidence from the outcome of the proposed approach, as shown in Figure 23.

The second case study, namely source "LNK" of the Windows operating system, is analysed to provide relevant information related to the files that are frequently accessed by the user. Figures 25 and 26 show the outcome of the proposed approach in the form of four levels of the timeline of events and artefacts, corresponding to source "LNK". In fact, 6, 7, 10 and 18 different fields are considered at Events: high level, Events: low level, Artefact location: high level and Artefacts location: low level, respectively, along with the implementation of distinct mechanisms. The first two levels, i.e., Events: high level and Events: low level of the proposed approach, provides an overview of information related to the file that is used by a user many times, such as name (chrome history window 7), type (.txt) and location (D:\Doctorate Studies\) of a particular file. Moreover, the last two levels of timeline, i.e., Artefact location: high level and Artefact location: low level, provide additional detailed and vital information related to a particular file that is frequently used by a user. It includes the type of operation performed by a user with that file, such as modified, accessed, changed or created. One can easily understand the timeline related to a particular file frequently accessed by a user and analyse the information to obtain digital evidence from the outcome of the proposed approach, as shown in Figure 26.

4.2.2.2. Android Operating System

The two case studies of Android operating system based timelines, namely source "META" and "FILE", are considered to show the capability of a novel approach. Similarly, like Windows operating system for the Android operating system, the same number of fields is considered at each level of approach, and the same mechanisms are stipulated to compose a relevant timeline. In fact, 6, 7, 10 and 18 different fields are considered at Events: high level, Events: low level, Artefact location: high level and Artefacts location: low level, respectively, along with the implementation of distinct mechanisms at each level to address other issues in the timeline. The first study case, i.e., source "META", is analysed to provide information related to different actions performed by the user related to the files and folders, such as Modification, Access, Change and Birth (created). Figures 27 and 28 present the outcome of the proposed approach in the form of four levels of abstraction approach, corresponding to the source "META". The first level of approach, i.e., Events: high level, provides brief information related to particular action performed by the user related to a file and folder.

Events: high level (level 1)
Date: 11/10/2015
Time: 12:46:00
Source: META
Short: OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Reference: 77

Events: low level (level 2)
Date: 11/10/2015
Time: 12:46:00
Source: META
Short: OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Extra: number_of_paragraphs:25 total_time:0
Reference: 77

Artefact location: high level (level 3)
Date: 11/10/2015
Time: 12:46:00
MACB: M..B
Source: META
Source type: Open XML Metadata
Short: OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Extra: number_of_paragraphs:25 total_time:0
Desc: Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89
Reference: 77

Artefact location: low level (level 4)
Date: 11/10/2015
Time: 12:46:00
Timezone: UTC
MACB: M..B
Source: META
Source type: Open XML Metadata
Type: Content Modification Time; Creation Time
User & Host: -
Short: Author: User
Desc: Creating App: Microsoft Office Word App version: 14.0000 Last saved by: Vartotojas Author: User Revision number: 2 Template: Normal Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89 Hyperlinks changed: false Links up to date: false Scale crop: false
Version: 2
Filename: OS:D:\yu\Download\Using the marketing mix to drive change.docx
Inode & Notes: -
Format: Openxml
Extra: doc_security: 0; i4: 1; number_of_paragraphs: 25; sha256_hash: cd2d4ad6058b86d15c6fffcdb08cdd94deacdba41d7dc0397553ae0649f6aa59; shared_doc: false; total_time: 0
Reference: 77

Figure 27. The output of META source of Android operating system [17]

Keys:
Address and Name:
OS:D:\yu\Download\Using the marketing mix to drive change.docx
Operation type: M..B
Application: Microsoft Office Word
Description: Creating App: Microsoft Office Word App version: 14.0000 Last saved by: Vartotojas Author: User Revision number: 2 Template: Normal Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89

Figure 28. The output of META source of Android operating system [17]

Events: high Level (level 1)
Date: 12/07/2015
Time: 15:26:13
Source: FILE
Short:
D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Visit: -
Reference: 86

Events: low level (level 2)
Date: 12/07/2015
Time: 15:26:13
Source: FILE
Short:
D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Visit: -
Extra: file_size: 2167277; file_system_type: OS; is_allocated: True
Reference: 86

Artefact location: high level (level 3)
Date: 12/07/2015
Time: 15:26:13
MACB: M...
Source: FILE
Source type: OS Content Modification Time
Short:
D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Visit: -
Extra:
file_size: 2167277;
file_system_type: OS;
is_allocated: True
Desc:
OS:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Type: file
Reference: 86

Artefact location: low level (level 4)
Date: 12/07/2015
Time: 15:26:13
Timezone: UTC
MACB: M...
Source: FILE
Source type: OS Content Modification Time
Type: Content Modification Time
User: -
Host: -
Short:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Desc: OS:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3 Type: file
Version: 2
Filename: OS:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Inode: -
Notes: -
Format: filestat
Extra: file_size: 2167277; file_system_type: OS; is_allocated: True;
sha256_hash:
ae44d66a82fffb5a828ef83432eb147f564050355df62fe3ec6ba9cfa6908862
Reference: 86

Figure 29. The output of FILE source of Android operating system [17]

Keys:
Address and Name:
D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Operation type: M...
Application: SHAREit

Figure 30. The output of FILE source of Android operating system [17]

It includes the name, type and storage location of a particular file. The Events: low level and Artefact location: high level provide additional information. It includes what kind of operations are performed by a user with a specific file or folder, the contents of a file, such as number of words, number of characters and many more along with the information that is provided at Events: high level. Finally, Artefact location: low level provides more detailed and useful information, such as the author's name and the name of the application used to perform a particular operation with a file. Regarding the outcome of the proposed approach, one can easily understand the timeline and analyse relevant information from the timeline related to different actions or operations performed by a user with a particular file by using specific applications and many more, as shown in Figure 27.

The second case study, namely source "FILE" of the Android operating system, is analysed to provide relevant information related to files that are available in a particular Android operating system mobile device. Figures 29 and 30 show the outcome of the proposed approach in the form of four levels of timeline of events and artefacts. The same number of fields is considered at each level along with the stipulation of mechanisms to compose relevant information as considered in the "META" source. In this study case, the information about a particular file is provided at a different level of timeline with a distinct level of abstraction of detail. The first two-levels of approaches provide brief information related to a particular available file, such as name, type, location and size of the file. In this case, the name, type, location and size of the file are Tum_Ho_Mera_Pyar-K_K(DesiTape.Com), mp3(audio), D:\yu\SHAREit\audios\ and about 2.16 MB, respectively. The last two levels of approach provide more detailed information along with the information provided at the first two levels of approach. It includes the type of the operation performed by the user or the system itself to a file, an application used to perform the operation and many more. Considering the outcome of the proposed approach, one can easily understand the timeline and analyse relevant information from the timeline related to a specific file available on the Android operating system based mobile devices, such as name, type, location, size of the file, performed operation with a file and many more, as shown in Figure 29.

4.2.2.3. iOS Operating System

The two case studies of the iOS operating system timelines are discussed, i.e., source "WEBHSIT" and "iMessage". Like Windows and Android operating systems, the same number of fields is considered at each level of approach, and the same mechanisms are stipulated to compose a relevant timeline. The first case study, i.e., source "WEBHIST", is analysed to provide relevant information to a web activity performed by a user on the web. The first two levels of approach provide a higher level of abstraction of information related to the web activity. It includes the address of a specific web page accessed by the user, how this specific web page is accessed either by user's typed address in the address bar or clicked link and the content search by the user. The last two levels of the timeline of the proposed approach provide detailed information along with information provided at the

first two levels.

Events: high level (level 1)
Date: 10/31/2019
Time: 08:16:41
Source: WEBHIST
Short: https://www.kaunokolegija.lt
Visit: LINK 1
Reference: 2024

Events: low level (level 2)
Date: 10/31/2019
Time: 08:16:41
Source: WEBHIST
Short: https://www.kaunokolegija.lt
Visit: LINK 1
Extra: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Reference: 2024

Artefact location: high level (level 3)
Date: 10/31/2019
Time: 08:16:41
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Short: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Visit: LINK 1
Extra: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Desc: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies) [count: 1] Host: www.kaunokolegija.lt Visit Source: [SOURCE_SYNCED] Type: [LINK - User clicked a link] (type count 1 time)
Reference: 2024

Artefact location: low level (level 4)
Date: 10/31/2019
Time: 08:16:41
Timezone: UTC
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Type: Last Visited Time
User & Host: -
Short: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Desc: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies) [count: 1] Host: www.kaunokolegija.lt Visit Source: [SOURCE_SYNCED] Type: [LINK - User clicked a link] (type count 1 time)
Version: 2
Filename: OS:D:\applebackup\acf4b9617ef493f11fa0dd4e11bce6cd6eb5b3f2\fa\faf971ce92c3ac508c018dce1bef2a8b8e9838f1
Inode & Notes: -
Format: sqlite/chrome_history
Extra: schema_match: False; sha256_hash: 1ec938e2eed7efe16719dde363cf12efd3fc7201e1c995f54ce8d18fb55c497b
Reference: 2024

Figure 31. The output of WEBHIST source of iOS operating system [17]

Keys:
URL:
https://www.kaunokolegija.lt
Search Term: Kaunas College - modern and practical studies
Browser: Google Chrome
Description: : LINK - User clicked a link

Figure 32. The output of WEBHIST source of iOS operating system [17]

Events: high level (level 1)
Date: 18/01/2020
Time: 09:45:17
Source: iMessage
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Reference: 39925

Events: low level (level 2)
Date: 18/01/2020
Time: 09:45:17
Source: iMessage
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Extra: -
Reference: 39925

Artefact location: high level (level 3)
Date: 18/01/2020
Time: 09:45:17
MACB: ...B
Source: iMessage
Source type:
Apple iMessage Application
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Extra:-
Desc: iMessage ID: +370xxxxxxx4 Read Receipt: True Message Type: Received Service: SMS Message Content: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Reference: 39925

Artefact location: low level (level 4)
Date: 18/01/2020
Time: 09:45:17
Timezone: UTC
MACB: ...B
Source: iMessage
Source type: Apple iMessage Application
Type: Creation Time
User & Host: -
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Desc: iMessage ID: +370xxxxxxx4 Read Receipt: True Message Type: Received Service: SMS Message Content: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Version: 2
Filename:
OS:D:\applebackup\acf4b9617ef493f11fa0dd4e11bce6cd6eb5b3f2\3d\3d0d7e5fb2ce288813306e4d4636395e047a3d28
Inode & Notes: -
Format: sqlite/imessage
Extra: schema_match: False; sha256_hash: 78570d1699f93d2ccd80cdf525568b6133aa8084e153dac1c774d34d8cf8fc1e
Reference: 39925

Figure 33. The output of iMessage source of iOS operating system [17]

Keys:
iMessage: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Application: Apple iMessage

Figure 34. The output of iMessage source of iOS operating system [17]

It includes the name of the web browser used by the user to perform the operation on the web, the type of web operation performed by the user and many more. Considering the outcome of the proposed approach, one can easily understand the timeline and analyse relevant information from the timeline related to a specific web activity performed by a user. It includes the address of the web page accessed by a user, name of the web browser used by a user to browse a particular web page, content search on the web page and many more, as shown in Figures 31 and 32.

The second case study, namely source "iMessage" of the iOS operating system base mobile device, is analysed to provide relevant information related to an activity that is performed locally without the usage of the internet, as shown in Figures 33 and 34. The first two levels of the novel approach show the actual information that is shared between two users. Moreover, the last two levels of the approach provide vital and detailed information related to this activity. It includes the shared information, the receiver and sender of shared information, the contact number used to send or receive the information and much more. Considering the outcome of the proposed approach, one can easily understand the timeline and analyse relevant information from the timeline related to a specific local activity performed by a user on the iOS operating system based device. It includes the actual communication that has taken place between two users, the contact number of a user used in the communication and many more, as shown in Figure 33.

4.2.2.4. Comparison

This section is devoted to presenting the comparison studies of a novel proposed approach and the existing approaches for the reconstruction and analysis of timeline developed by different researchers and authors. The main idea behind the demonstration of presenting the comparative studies is to show that the proposed abstraction based approach is capable and helps analysing the timeline efficiently, as compared to the available approaches based on various factors or features, as shown in Table 10.

A comparative study is analysed by considering 21 different factors or features between the proposed approach and five existing approaches, i.e., Guðjónsson [59], Inglot and Liu [70], Soltani, Seno and Yazdi [124], Esposito and Peterson [47] and Hargreaves and Patterson [62]. These features illustrate that the timeline reconstructed by a particular approach consists of relevant information related to different operations or actions performed by a user on specific digital devices, the integrity of information, information related to the available applications, files and system files, extraction of mail addresses used by the user for communication or other purposes, the heterogeneity of operating system and many more, as shown in Table 10. It has been found from the obtained results (see Table 10) that the proposed abstraction based approach is capable of analysing the timeline more effectively and appropriately than the existing approaches. The proposed approach accomplishes all (21) features. However, the existing approaches do not have many of these features. Moreover, the proposed approach as well provides complete information related to all online and offline actions, available user and system files,

applications executed by the user, mail addresses and contact numbers used for the communication or other purposes and many more along with the stipulation of a unique structure at each level, elimination of duplicity of events and artefacts, exclusion of irrelevant information to compose a relevant and recognizable timeline by a user or digital investigators.

Table 10. Comparison of proposed abstraction based approach and the existing approaches

Features	Our approach	Guðjónsson [59]	Inglot and Liu [70]	Soltani, Seno and Yazdi [124]	Esposito and Peterson [47]	Hargreaves and Patterson [62]
1. The reference number used to keep a relationship with the original source file.	+	-	-	-	-	-
2. Elimination of duplicity and irrelevant details.	+	-	-	-	-	-
3. Splitting of timeline of events and artefacts.	+	-	-	-	-	-
4. Stipulation of unique structure at each level.	+	-	-	-	-	-
5. Extraction of MAC address.	+	-	-	-	-	-
6. Extraction of different applications used by a user and the system itself.	+	+	-	+	+	-
7. Extraction of mail addresses used for the communication.	+	-	-	+	+	-
8. Attaining information related to files and applications that are no longer available.	+	+	-	-	-	-
9. Attaining information related to supportive applications and systems files.	+	+	-	+	-	-
10. Retrieval of complete address of the web pages accessed by a user.	+	+	+	+	+	+
11. How (LINK, TYPED, FORM_SUBMIT, WEBHIST) specific web page is accessed.	+	-	+	+	-	-
12. Retrieval of actual information search by a user on the web.	+	+	+	+	+	+
13. List of different web browsers used by a user to perform web activities on the web.	+	+	-	+	+	+
14. Attaining of information related to the files or applications, which are downloaded	+	-	-	-	+	-

from the web.						
15. Retrieval of information related to the files and applications that are most frequently accessed by a user.	+	+	-	-	-	-
16. Information related to available all types of files and applications.	+	+	-	+	+	-
17. Information related to different types of operations performed by a user with specific file, i.e., MACB.	+	-	-	+	-	-
18. Retrieval of actual information (imessage) shared among the users.	+	-	-	-	-	-
19. Retrieval of contact number of users used for communication and other purposes.	+	-	-	-	-	-
20. Heterogeneity of operating systems.	+	+	-	-	-	-
21. Integrity of data.	+	+	-	+	+	-

4.3. Finding Novel Ontology

4.3.1. Experimental Setup

The novel developed ontology is based on the abstraction based approach for the analysis of the timeline and is implemented by using the ontology editor Protégé 5.5.0 Build Beta-9 version along with the visualization plugins, namely OWLViz, OntoGraf and VOWL, and the ontology visualization tool OWLGred [82] [17]. The fundamental concepts of the digital forensic domain are already explained in chapter 3 “RESEARCH DESIGN AND METHODS”. These fundamental concepts are defined by designing an ontology where the concepts and sub-concepts are explained by using classes, sub-classes and properties. There are various new terminologies that have been encountered during the analysis of the timeline of different operating systems based devices, namely Windows, Android and iOS. Thus, the novel ontology should as well contain information of these newly encountered terminologies. For this, new sub-classes are defined, corresponding to each newly encountered terminology under the sub-class Artefact_Reference of the proposed ontology.

4.3.2. Results

The outcomes show that eleven new terminologies are encountered during the analysis of the timeline of Windows, Android and iOS operating systems based devices. These terminologies are added in the proposed ontology by defining a new sub-class corresponding to each novel term under the Artefact_Reference subclass,

as shown in Figure 35. Moreover, Table 11 provides descriptive information of these newly encountered terminologies. It has been found from the obtained results that the Windows operating system based timeline contains a maximum number of new nine terms, namely “LNK”, “REG”, “OLECF”, “RECBIN”, “META”, “WEBHIST”, “LOG”, “PE” and “FILE”. The Android operating system based timeline contains four terminologies, namely OLECF”, “META”, “PE” and “FILE”. However, three terms, i.e., “OLECF”, “META” and “PE”, except the term “FILE”, provide the same information as described in Table 11. The two new subclasses “Application” and “Browsing_History” of the subclass “FILE” are explicitly defined to show information about browsing activities browsed by a user, as there is no information implicitly available in the timeline for the browsing activities [17].

The iOS operating system based timeline contains six new terminologies, namely “META”, “PE”, “WEBHIST”, “FILE”, “IMESSAGE” and “PLIST”. The four terms, except the two terms, namely “IMESSAGE” and “PLIST”, provide the same information, which is described in Table 11. The “IMESSAGE” terms provide information related to all received and send messages by using Apple iMessage Application on a particular iOS operating system based device. It shows crucial information, such as the contact number, which is used to send and receive messages. The “PLIST” stands for Property List and is a format for storing the application data. It was originally developed by Apple for iOS operating system based devices [17].

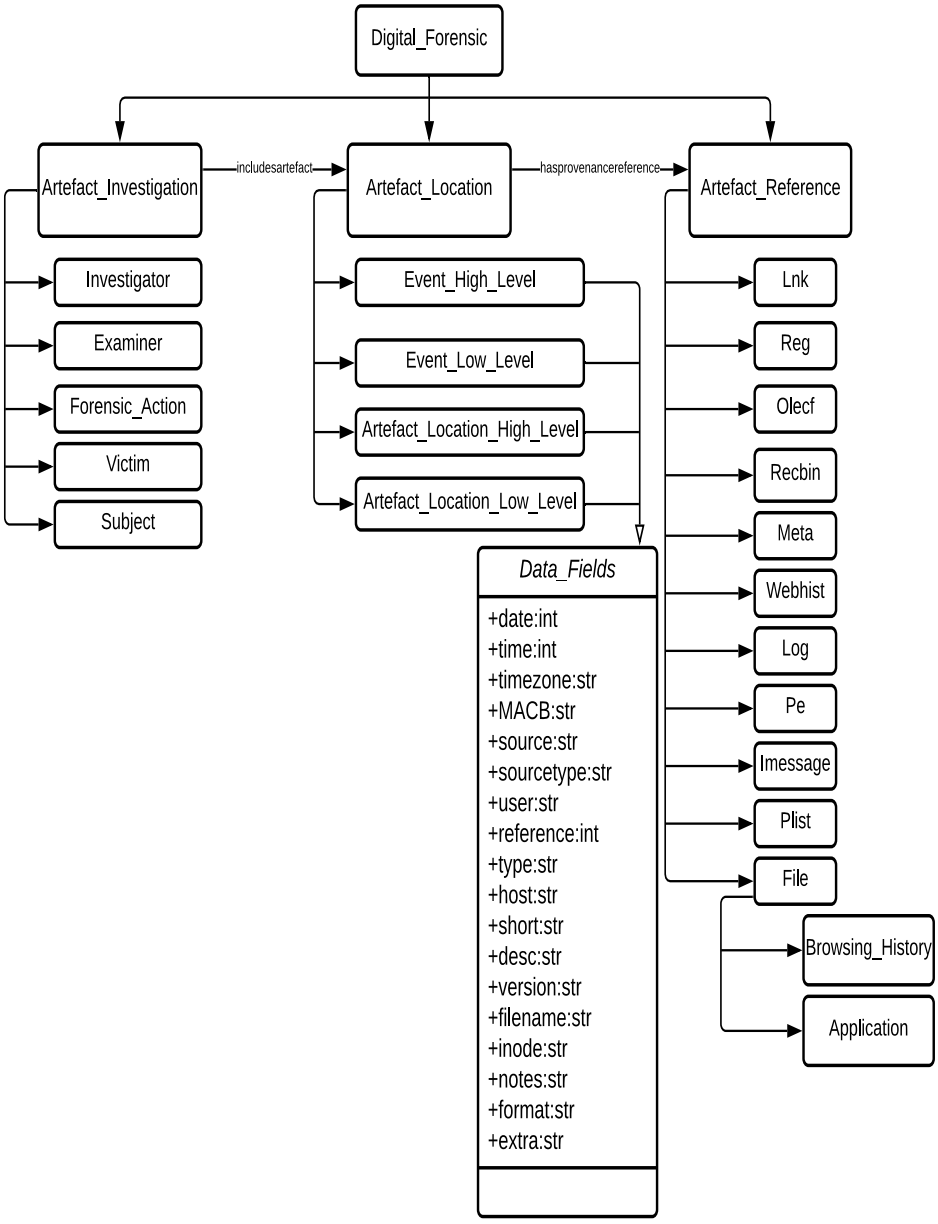


Figure 35. The proposed ontology

Additionally, an abstract class, i.e., “Data_Fields”, is as well defined in the proposed ontology, which contains eighteen data properties. These data properties are shared by the four sub-classes of sub-class, i.e., “Artefact_Location”, and are illustrated by using the inheritance relationship, as shown in Figure 35. Moreover, the “Artefact_Location” sub-class represents a novel abstraction based approach for

the timeline analysis, and its four sub-classes illustrate four levels of the timeline of events and artefacts of the approach, namely Events: high level, Events: low level, Artefact location: high level and Artefact location: low level. Thus, the four sub-classes, namely "Event_High_Level", "Event_Low_Level", "Artefact_Location_High_Level" and "Artefact_Location_Low_Level", inherit different data properties for the analysis of the timeline.

Table 11. New terminologies and their descriptions [17]

S.No.	**Terminologies (Sub-Classes)**	**Description**
1.	WEBHIST	Information related to browsing activities, it includes the addresses of web pages accessed by a user, the mail addresses of a user, downloaded files and application (name of the web browser) used to access online services.
2.	RECBIN	Information about files that have been deleted by a user from the system and recycle bin.
3.	PE	PE stands for Portable Executable. PE formatted files include: .exe, .apk, .ipa .dll and .sys (driver files).
4.	FILE	Information related to a particular file is provided, such as name, type, location and size.
5.	META	Meta often is described as "data about data". It includes Modification, Access, Change and Birth (created) information of a particular document or file.
6.	LNK	Shortcut files that are connected to an application or file commonly found on the desktop of a user or throughout a system and end with .LNK extension. It is very helpful to access files that are no longer available in the system.

7.	REG	Information about the used applications and .DAT files that are only meant for the support of applications.
8.	OLECF	OLECF stands for Object Linking and Embedding compound file. It contains .msp, .msi, .asd and .automaticdestination-ms files, which provide information about the updates of Windows operating system and other programs.
9.	LOG	Stores information about the events that take place in an operating system or other program runs.
10.	IMESSAGE	Provides information related to all received and sent messages by using the Apple iMessage Application.
11.	PLIST	Stands for Property List and is a format for storing application data.

Figure 36 illustrates the study case of source “WEBHIST” to demonstrate the developed ontology and an abstraction based approach for the timeline analysis together. The main idea of presenting a study case is to show the complete prototype of a developed ontology in a single scenario along with the position of different levels of events and artefacts of the approach. Thus, all fundamental and novel terminologies of the digital forensic domain and the relevance of the approach for the timeline analysis are well understood by the users or digital investigators. Moreover, the users can as well observe and understand the relationship between the novel ontology and abstraction approach for the timeline analysis precisely. The figure as well shows a different number of data properties inherited by different sub-classes of sub-class “Artefact_Location” for the analysis of timeline at different levels of events and artefacts.

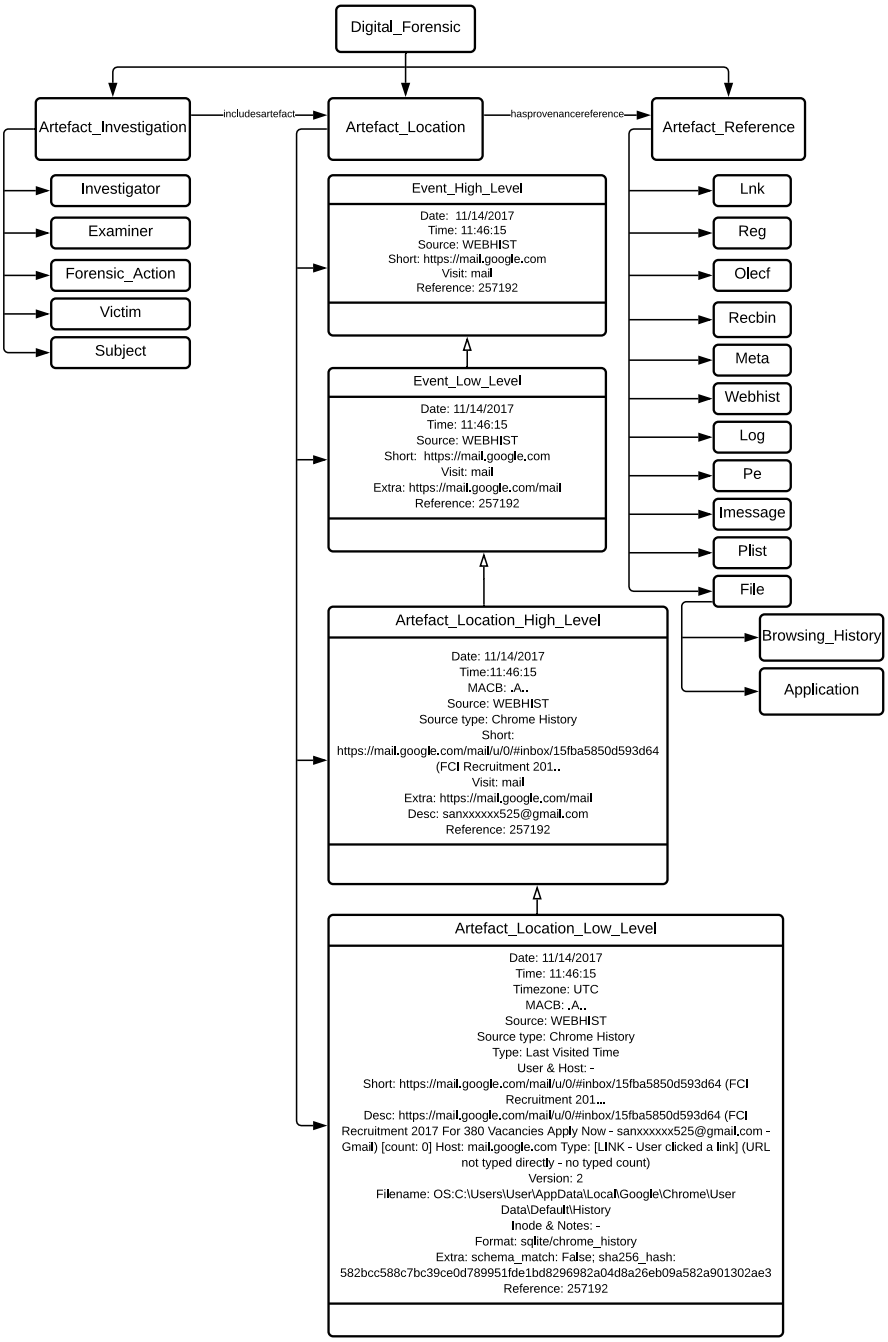


Figure 36. The novel ontology with abstraction approach [17]

The figure illustrates that a novel ontology contains a base class “Digital _Forensic” that represents the general concept in the forensics domain, for which ontology is being developed, to define common vocabulary, share information and reuse and analyse domain knowledge. The three sub-classes have derived from the base class, namely “Artefact_Investigation”, “Artefact_Location” and “Artefact_Reference”, to define various fundamental and new terminologies of the digital forensic domain. The first sub-class “Artefact_Investigation” defines the basic terms and distinct actions performed by different specialists of the digital forensics domain. The second class “Artefact_Location” illustrates the novel abstraction based approach for the timeline analysis, and it contains four sub-classes corresponding to four levels of approach. Additionally, the developed ontology as well shows that there are different numbers of data properties and numerous features inherited by distinct levels of approach for effective timeline analysis, as shown in Figure 36.

In this study, 6, 7, 10 and 18 data properties are inherited by Events: high level, Events: low level, Artefact location: high level and Artefact location: low level of approach, respectively, to assist the user in understanding the web activity performed by a user on the internet. It includes the date and time of specific web activity, the complete web address of the internet page accessed by a user, the title of content browsed by a user, the name of the web browser used to access a particular web page and many more. Moreover, all these features and information enables ontology to support digital investigators during their investigation to understand the newly encountered terminologies and their relevance. The third and the last sub-class “Artefact_Reference” shows that there are eleven different novel terminologies that are encountered during the analysis of the timeline of different operating systems based devices, namely Windows, Android and iOS, and their descriptions are available in Table 11.

4.3.3. Comparison

In this section, two comparisons between Windows, Android and iOS operating systems are discussed. First, the comparison shows the maximum number of new terminologies, which present the sources of information available for each operating system, the number of fields (data properties) and the number of sources (artefact references) are considered at different levels of approach to reconstruct the timeline [17].

Table 12. Comparison of Windows, Android and iOS operating systems [17]

Operating System	Artefact_Location	Number of Fields (Data Properties)	Number of Artefact_Reference
Windows Operating System	Event: high level	6	6
	Event: low level	7	9
	Artefact location: high level	10	9
	Artefact location: low level	18	9
Android Operating System	Event: high level	6	3
	Event: low level	7	4
	Artefact location: high level	10	4
	Artefact location: low level	18	4
iOS Operating System	Event: high level	6	3
	Event: low level	7	4
	Artefact location: high level	10	6
	Artefact location: low level	18	6

It has been found from the results (see Table 12) that 9, 4 and 6 are the maximum numbers of sources (artefact reference) available for Windows, Android and iOS operating systems, respectively. For all operating systems, 6, 7, 10 and 18 fields (data properties) for the Events: high level, Events: low level, Artefact location: low level and Artefact location: high level, respectively, are considered in the timeline. It shows a unique structure is followed at each level of abstraction based approach for the analysis of all operating systems based timelines.

The second comparison shows the availability of different types of information available in the timelines of these three operating systems. It has been observed from the results (see Table 13) that the timeline of Window-based system provides the maximum information regarding all types of operations executed by the user. It

includes online and offline, the most frequently used application and deleted files, as compared to Android and iOS operating systems [17].

Table 13. Comparison of Windows, Android and iOS operating systems [17]

Parameters	Windows Operating System	Android Operating System	iOS Operating System
Online Operation Information	Available	Partially available	Available
Offline Operation Information	Available	Not available	Available
Mail Addresses	Available	Not available	Available
Frequently Used Application	Available	Not available	Not available
MAC Address	Available	Not available	Not available
Deleted Files	Available	Not available	Not available

4.3.4. Evaluation of Novel Ontology

A novel ontology-based on abstraction based approach for the analysis of the timeline is evaluated by using two methods, namely ontology taxonomy evaluation and ontology content evaluation, for the verification and validation of novel ontology, respectively.

4.3.4.1. Verification of Novel Ontology

The ontology taxonomy evaluation method is used for the verification and provides the following results (see Table 14).

Table 14. Verification of novel ontology

1. Inconsistency	**I. Circularity error:** No error because no class in a novel ontology is illustrated as a specialization or generalization of itself.
	II. Partitions error: No partitions error found during the evaluation of the proposed novel ontology. The developed ontology does not contain an incorrect description of disjoint classes, i.e., a class of developed ontology is not a derived class of more than one class.
	III. Semantic error: The developed ontology does not consist of semantic error because all classes or concepts of ontology are classified semantically correct.
2. Incompleteness	**I. Incomplete concept classification:** The novel ontology contains all basic concepts of the digital forensic domain. The ontology as well has terminologies of abstraction approach for the timeline analysis and new terminologies corresponding to distinct operating systems (Windows, Android and iOS) based devices.
	II. Partition error: The developed ontology does not contain partition error.
3. Redundancy	**I. Grammatical redundancy:** As no class in the developed ontology has more than one definition or description; thus, the novel ontology does not have grammatical redundancy errors.
	II. Identical formal definition of some classes and instances: The developed ontology does not consist of identical formal definitions of some classes and instances error because the novel ontology does not have the same definition corresponding to more than one class and instances.

4.3.4.2. Validation of Novel Ontology

The ontology content evaluation method is used for the validation and provides the following results (see Table 15).

Table 15. Validation of novel ontology

1. Consistency	The novel ontology is consistent as each definition of a class is consistent, according to the real world, and no contradictory information can be obtained from the definition.
2. Completeness	The developed ontology contains all basic terminologies of the digital forensic domain along with the novel terminologies corresponding to the distinct operating system based devices and terminologies corresponding to the abstraction approach for the timeline analysis. Moreover, the ontology as well contains definitions of each concept of a class. Thus, the proposed ontology is a complete ontology.
3. Conciseness	The proposed ontology is concise because it does not contain irrelevant definitions of classes or concepts along with the repetition of definitions.
4. Expandability	The new classes or concepts and definitions corresponding to them can be easily added without changing the already well-defined definitions in the developed ontology. Thus, it means that the development is easily expandable. For instance, a new class and the definition corresponding to it can be easily added under the sub-class "Artefact_Reference" of a novel ontology without altering other concepts.
5. Sensitiveness	As the novel ontology is easily expandable, it means that the minor changes in some definitions will not change all well-defined classes or the concept. Thus, the developed ontology is not sensitive.

The outcome of both methods, namely ontology taxonomy evaluation and ontology content evaluation for the verification and validation of ontology, illustrate that the novel ontology-based on the abstraction approach for the timeline analysis achieves all factors. These factors are as follows: inconsistency, incompleteness, redundancy, consistency, completeness, conciseness, expandability and sensitiveness that are required for the verified and validated ontology.

4.4. Finding Visualization

4.4.1 Experimental Setup

In this section, the visualization of the timeline, analysed by a novel abstraction based approach, is explained. The visualization is implemented on the outcome of a novel abstraction based approach by using the software Gephi version 0.9.2 201709241107. Gephi is an open-source software for the network or graph visualization and analysis. Gephi assists to intuitively find patterns and trends,

highlight outliers and tells stories with their data. In order to present huge graphs in real-time and enhance the pace of exploration, it uses 3D render engine. Gephi comes with many included functionalities and flexible architecture to explore, analyse, spatialize, filter, cluster, manipulate and export various types of networks [120].

The outcome of the novel abstraction based approach for the analysis of the timeline is in the form of textual information, as shown in chapter 3. Thus, the output of a novel approach, corresponding to numerous and different activities performed by the user on a specific digital device, is not possible to depict in a single figure at the same time in a relevant way. For this, the visualization is implemented on the outcome of a novel approach to illustrate the timeline corresponding to numerous activities graphically in a figure. Moreover, the visualization as well facilitates to display information in a figure to users in such a way that allows them to interpret the underlying information related to multiple activities performed by a user at a single period of time, and this is the objective of the implementation of visualization in a research work.

4.4.2. Results

A case study of the timeline, analysed by the abstraction based approach, namely "WEBHIST" source of Windows operating system, is examined by using the graph-based visualization technique. The main idea of using the visualization technique is to demonstrate the benefits of visualization to facilitate the information related to many activities carried out by a user in a recognizable and visual way, which is hard to achieve in text format data or information. Moreover, the human brain can interpret information easily and quickly from the visual cues, as compared to the written language.

```
11/13/2017;16:04:26;.A..;WEBHIST;Firefox History;URL:
http://go.microsoft.com/fwlink/p/?LinkId=255141;TYPED 1;"extra: [u'(URL
not typed directly)'  u'Transition: TYPED'];
";http://go.microsoft.com/fwlink/p/?LinkId=255141 [count: 1] Host:
go.microsoft.com (URL not typed directly) Transition: TYPED;252990
11/13/2017;16:04:34;.A..;WEBHIST;Firefox History;URL:
http://google.com/;TYPED 2;"extra: [u'(URL not typed directly)'
u'Transition: TYPED']; ";http://google.com/ [count: 2] Host: google.com
(URL not typed directly) Transition: TYPED;253102
11/13/2017;19:01:19;.A..;WEBHIST;Chrome
History;https://www.magnetforensics.com/computer-forensics/how-to-
analyze-usb-device-...;LINK ;https://www.magnetforensics.com/computer-
forensics ;https://www.magnetforensics.com/computer-forensics/how-to-
analyze-usb-device-history-in-windows/ (How to Analyze USB Device History
in Windows - Magnet Forensics Inc.) [count: 0] Host:
www.magnetforensics.com Type: [LINK - User clicked a link] (URL not typed
directly - no typed count);255696
11/13/2017;19:02:22;.A..;WEBHIST;Chrome
History;https://www.magnetforensics.com/digital-forensics-
software/internet-evidence-...;LINK
;https://www.magnetforensics.com/digital-forensics-software
;https://www.magnetforensics.com/digital-forensics-software/internet-
evidence-finder (Magnet IEF) [count: 0] Host: www.magnetforensics.com
Visit from: https://www.magnetforensics.com/computer-forensics/how-to-
analyze-usb-device-history-in-windows/ (How to Analyze USB Device History
in Windows - Magnet Forensics Inc.) Type: [LINK - User clicked a link]
(URL not typed directly - no typed count);255719
11/13/2017;19:02:30;.A..;WEBHIST;Chrome
History;https://www.magnetforensics.com/try-internet-evidence-finder-
free-for-30-days...;LINK ;https://www.magnetforensics.com/try-internet-
evidence-finder-free-for-30-days... ;https://www.magnetforensics.com/try-
internet-evidence-finder-free-for-30-days/ (Try Magnet IEF for Digital
Forensics Free for 30 Days - Magnet Forensics Inc.) [count: 0] Host:
www.magnetforensics.com Visit from:
https://www.magnetforensics.com/magnet-ief/ (Magnet IEF) Type: [LINK -
User clicked a link] (URL not typed directly - no typed count);255724
```

Figure 37. The L2TCSV format

Figure 37 illustrates the outcome of the proposed abstraction based approach for the timeline analysis in L2TCSV (Log2timeline comma-separated values) format, which is textual information related to different activities performed by a user on the web. A user can view very few events or activities from this figure at a particular period of time and require more time to interpret the information. For this, the graph-based visualization technique is implemented on all four levels of the timeline of abstraction based approach corresponding to “WEBHIST” source of Windows operating system.

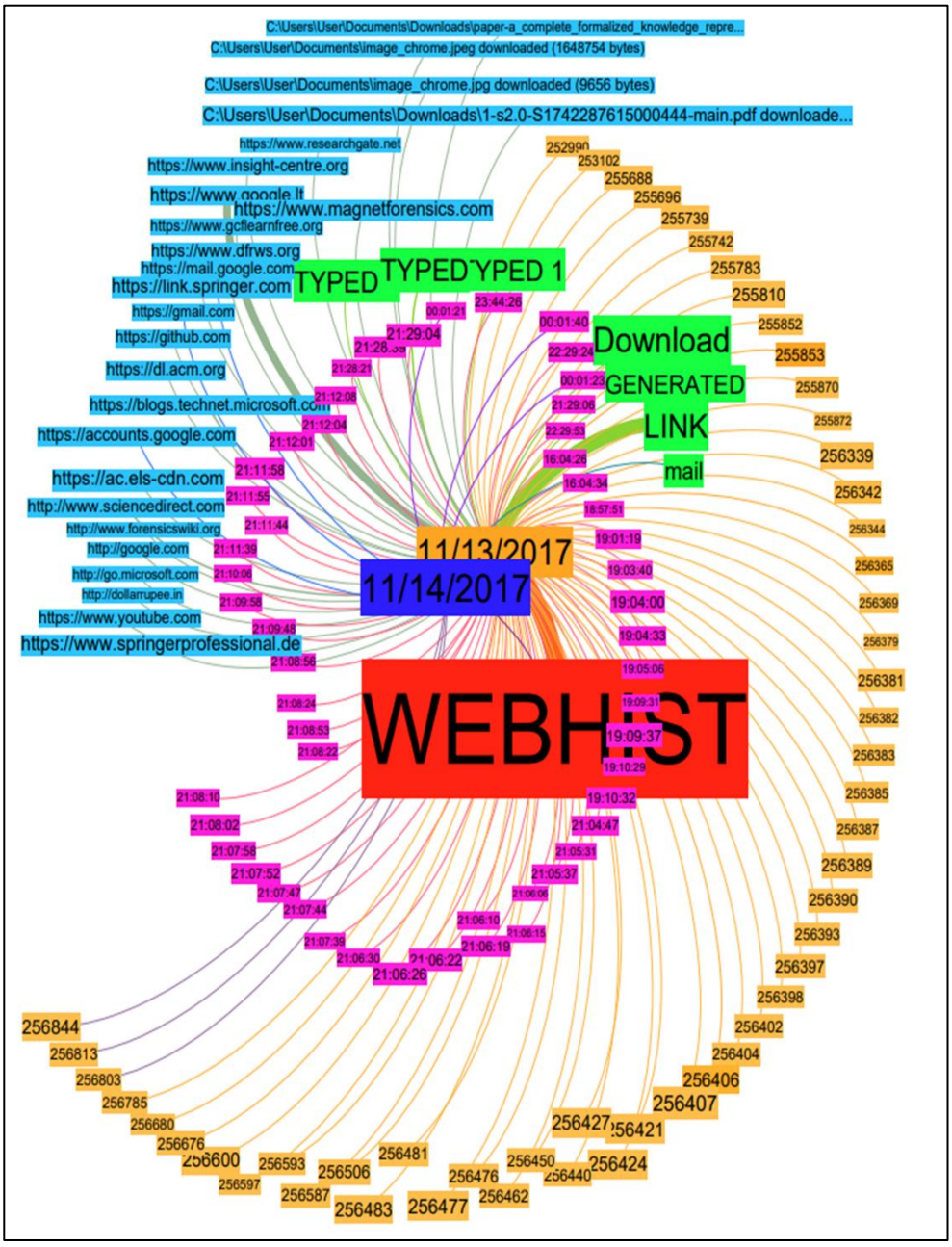


Figure 38. Event high level visualization

Figure 38 shows the visualization of Events: high level of abstraction based approach for the timeline analysis corresponding to “WEBHIST” source. One can easily and instantly view and interpret a large volume of information from the results of graph-based visualization than the textual information. The different sizes and colours of polygon-shaped nodes and edges along with dual circle layouts features are utilized to compose information visually and understandably. In case of this study, the visualization corresponding to Events: high level very clearly shows the different actions performed by a user on the internet or the web along with the information related to these actions. A user can quickly view and interpret the information, such as the names of web pages accessed by a user, storage locations of information downloaded from the web, what kind of operations are performed by a user on the web, date and time of actions and many more, which required more time in textual information, as compared to the graph-based visualization. For instance, the name of the accessed web pages includes: http://go.microsoft.com, https://www.magnetforensics, http://www.forensicswiki.org, http://www.sciencedirect.com; the storage location of download information from the web includes: C:\Users\User\Documents\image_chrome.jpg downloaded (9656 bytes), :\Users\User\Documents\image_chrome.jpeg downloaded (1648754 bytes), and many more are clearly visible and represented by blue polygon-shaped nodes. Similarly, Figure 39 as well illustrates the visualization of Events: low level of abstraction based approach for the timeline analysis corresponding to “WEBHIST” source. The outcomes of graph-based visualization show very clear and understandable information with more details. It includes the complete addresses of the web pages accessed by a user, which provide more perspective for understanding the exact information search on the web, such as https://www.researchgate.net/publication,https://www.obitko.com/tutorials, https://moodle.ktu.edu/?lang=en(MOODLE.KTU.EDU, https://www.google.lt/search?dcr=0&ei=FwkKWvOGJ4aUgAb0uZ3IBA&q=ontology+based, and many more demonstrated by red polygon-shaped nodes.

The graph-based visualization of Artefact location: high level of abstraction based approach for the timeline analysis corresponding to “WEBHIST” source is shown in Figure 40. The visualization at this level provides more detailed and meaningful information related to various actions performed by a user on the web along with the information provided by the graph-based visualization of the previous two levels of abstraction based approach, i.e., Events: low level and Events: high level. For instance, the name of the web browser used by a user to perform distinct operations on the web and what kind of operations performed by a user, which is illustrated by MACB, i.e., modified, accessed, change and create (birth). In this study case, the two different web browsers are used by a user to perform various activities on the web, i.e., Google Chrome and Mozilla Firefox, and A, i.e., access operation, is frequently performed by the user.

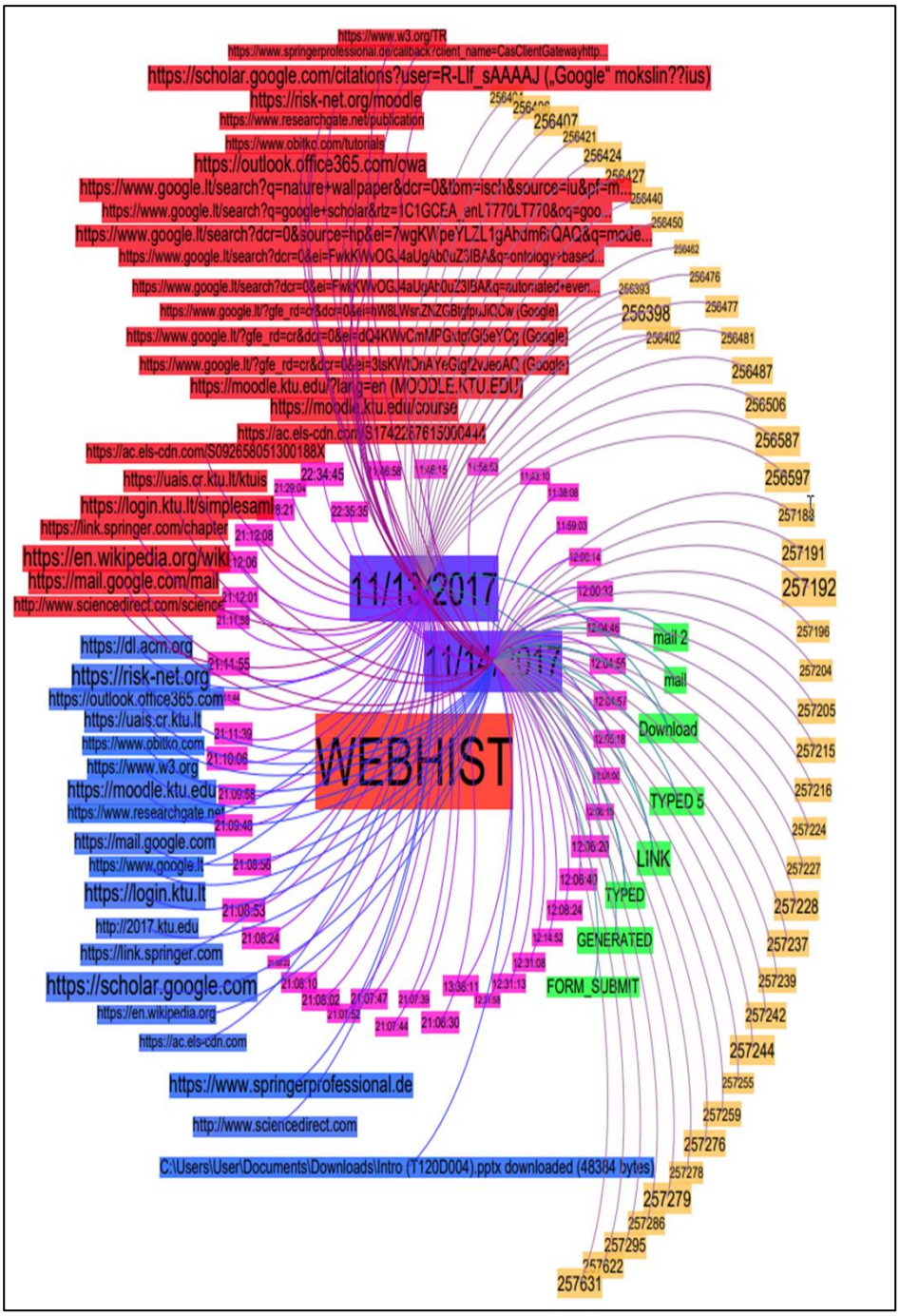


Figure 39. Event: low level visualization

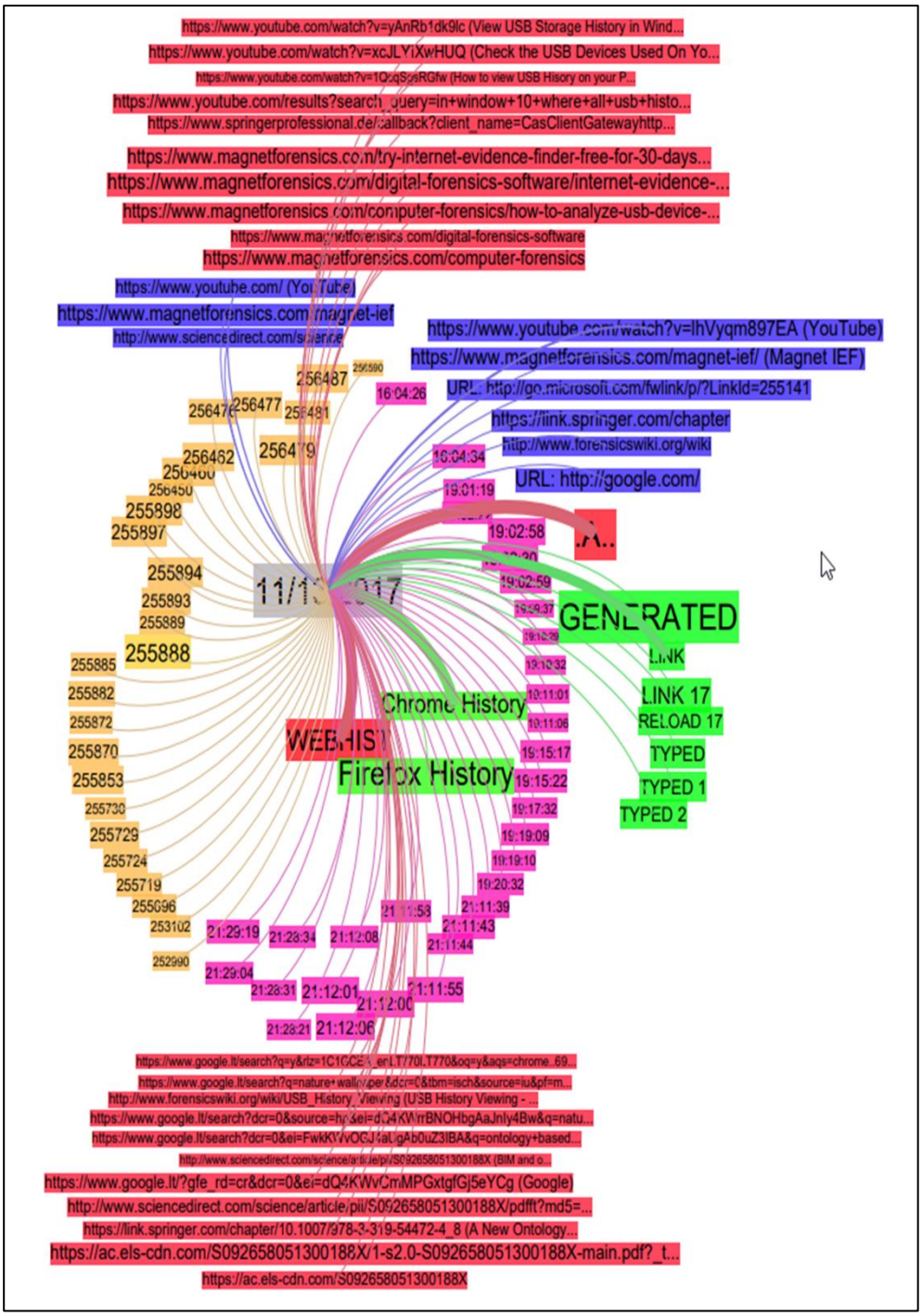


Figure 40. Artefact location: high level visualization

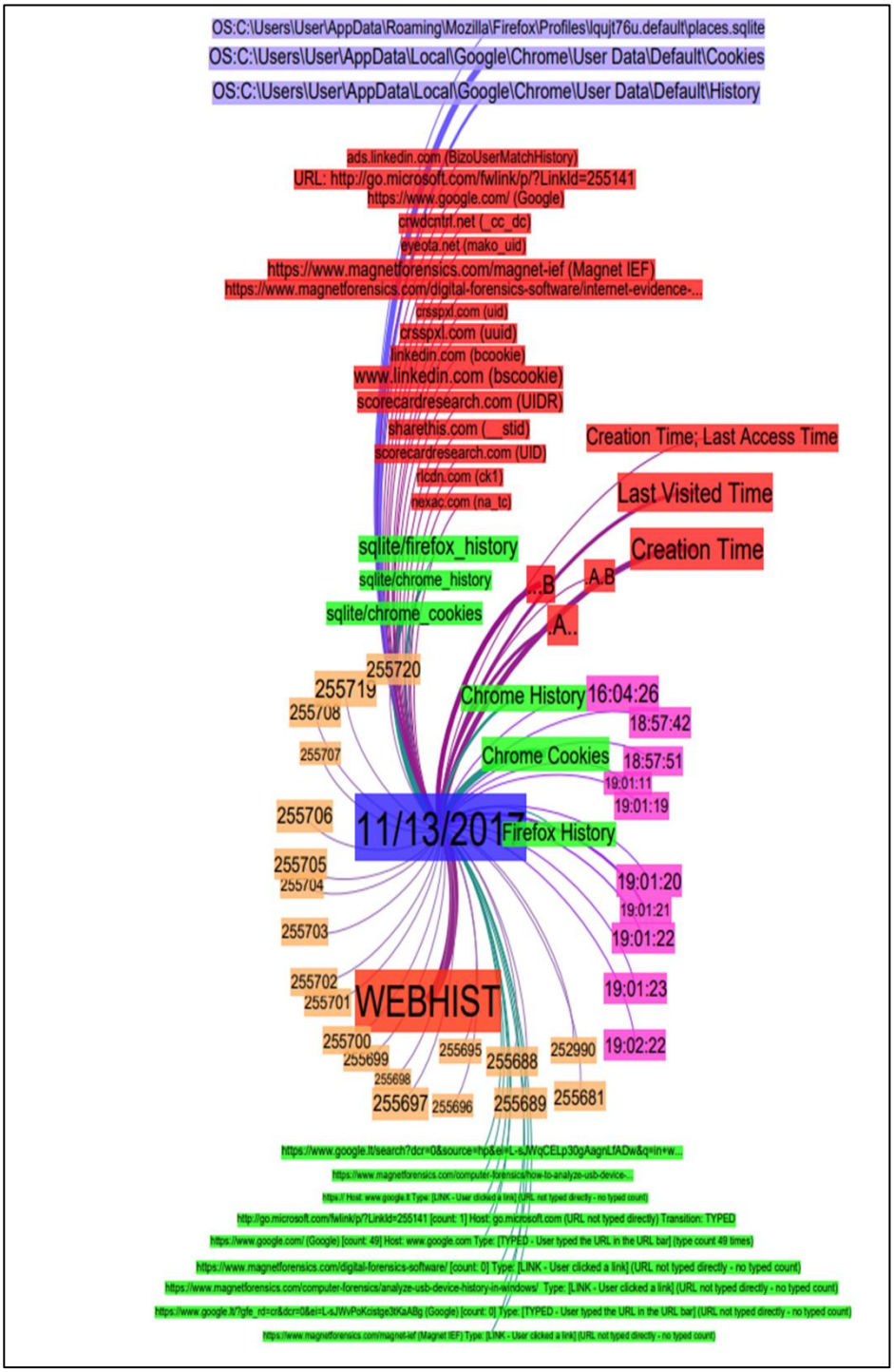


Figure 41. Artefact location: low level visualization

These web browsers and operations (A, i.e., access) are clearly illustrated in the graph-based visualization (see Figure 40) of Artefact location: high level by green polygon-shaped nodes and red polygon-shaped nodes, respectively.

Figure 41 shows the graph-based visualization outcome of Artefact location: low level of abstraction based approach for the timeline analysis corresponding to "WEBHIST" source. The graph-based visualization at this level provides more detailed and clear information than the previous three levels of the abstraction based approach. It includes the addresses of the web pages accessed by a user along with detailed information on how these web pages are accessed by a user. For instance, https://www.google.com/ (Google) [count: 49] Host: www.google.com Type: [TYPED - User typed the URL in the URL bar] (type count 49 times), https://www.magnetforensics.com/magnet-ief (Magnet IEF) Type: [LINK - User clicked a link] (URL not typed directly - no typed count), https://www.magnetforensics.com/computer-forensics/analyze-usb-device-history-in-windows/ Type: [LINK - User clicked a link] (URL not typed directly - no typed count), https:// Host: www.google.lt Type: [LINK - User clicked a link] (URL not typed directly - no typed count) and many more. This information is clearly visible and interpretable and shown by the green polygon-shaped nodes. Moreover, more information related to MACB operations is as well available at this level, such as creation time, visited time and access time, and clearly shown by red polygon-shaped nodes. Further, this level as well provides information related to the database used by browsers to store their internal data, such as browser history. The graph-based visualization clearly shows that Google Chrome and Mozilla Firefox use SQLite database to store their internal data and are presented by green polygon-shaped nodes.

Moreover, Table 16 is as well included to illustrate the complete overview of visualization of information at different levels of abstraction based approach by using graph-based visualization technique, corresponding to WEBHIST source. This will assist the user to select a particular visualization among the available visualizations corresponding to four levels of timeline of abstraction based approach, according to his/her requirements, to visualize the information and understand the timeline more precisely.

Table 16. Graph-based visualization

Graph-based visualization technique				
Visualized information	**Events: high level**	**Events: low level**	**Artefact location: high level**	**Artefact location: low level**
Name of event or action (WEBHIST)	✓	✓	✓	✓

Date and time	✓	✓	✓	✓
Address of the web page (abstract)	✓	✓	✓	✓
Reference number	✓	✓	✓	✓
Type of actions (download, mail, LINK)	✓	✓	✓	✓
Storage location	✓	✓	✓	✓
Address of the web page (detailed)	×	✓	✓	✓
Web browser information	×	×	✓	✓
Operation's information (MACB)	×	×	✓	✓
Additional information related to the performed actions (LINK, TYPED)	×	×	×	✓
Additional information related to the used web browsers	×	×	×	✓

4.5. Summary

The capability of the proposed abstraction based approach for the analysis of the timeline is demonstrated by implementing it on Windows, Android and iOS operating systems. The results of two case studies from each operating system are discussed and examined to show the output of the novel approach. The outcome of the approach illustrates that the novel approach is able to reduce the complexity of

the timeline by analysing and splitting the timeline into four levels of abstraction of the timeline of events and artefacts. Moreover, it has been found from the results that at each level of approach, different sources and fields are examined for analysing the timeline to omit unwanted details, enforcing the correctness of the timeline and presenting only information that will be helpful to recognise and understand particular actions performed by a user. A comparative study is as well performed to show that the abstraction based approach is able to analyse the timeline efficiently as compared to the available approaches based on various factors or features. Five different available approaches for the analysis of the timeline and twenty-one distinct features are considered in the comparative study. It has been found from the outcomes of comparative studies that the novel abstraction based approach is able to analyse the timeline efficiently. The proposed approach accomplishes all (21) features. However, the existing approaches do not have many of these features.

The fundamental concepts of the digital forensic domain are already explained in chapter 3. These fundamental concepts are defined by designing an ontology where the concepts and sub-concepts are explained by using classes, sub-classes and properties. In this chapter "Finding of novel ontology", new sub-classes are defined corresponding to each newly encountered terminologies under the sub-class "Artefact_Reference" of the proposed ontology along with the description of new terminologies, as shown in Table 9. It has been found from the results that there are eleven new terminologies that have been encountered during the analysis of the timeline of Windows, Android and iOS operating systems based devices. The Windows operating system based timeline contains a maximum number of new nine terms, namely "LNK", "REG", "OLECF", "RECBIN", "META", "WEBHIST", "LOG", "PE" and "FILE". The Android operating system based timeline contains four terminologies, namely "OLECF", "META", "PE" and "FILE". However, three terms, i.e., "OLECF", "META" and "PE", except the term "FILE", provide the same information as Windows operating system. The two new subclasses "Application" and "Browsing_History" of the subclass "FILE" are explicitly defined to show information about the browsing activities browsed by a user, as there is no information implicitly available in the timeline for browsing activities. The iOS operating system based timeline contains six new terminologies, namely "META", "PE", "WEBHIST", "FILE", "IMESSAGE" and "PLIST". The four terms, except the two terms, namely "IMESSAGE" and "PLIST", provide the same information as the other two operating systems. Moreover, a case study of source "WEBHIST" is as well discussed to show the complete prototype of a developed ontology in a single scenario along with the position of different levels of events and artefacts of approach. Thus, all fundamental and novel terminologies of the digital forensic domain and the relevance of the approach for the timeline analysis are well understood by the users or digital investigators. The developed ontology based on the abstraction approach for the timeline analysis is verified and validated by using ontology taxonomy evaluation and ontology content evaluation methods, respectively. The outcomes of both methods illustrate that the novel ontology

accomplishes all factors of these two methods. It means that the developed ontology is complete, consistent, non-redundant, concise, expandable and sensitive. Thus, the novel ontology is technically verified and validated.

The output of the proposed approach of the analysis of timeline is in the textual form. Thus, it is not possible to show the outcome of the proposed approach, corresponding to many activities in one figure at the same time, in a relevant way. In order to address this issue, the graph based visualization is implemented on the outcome of a novel approach to illustrate the outcome of the approach corresponding to numerous activities graphically in a figure. A case study of graph based visualization is discussed to show the benefits of visualization of facilitating the information related to many activities carried out by a user at the same period of time in a single figure in a recognizable way. It has been found from the results of the implementation of visualization on a novel approach that the user is able to attain and interpret the information related to multiple activities carried out by a user on a specific digital device at the same unit of time from the single figure of outcome precisely.

5. CONCLUSIONS

1. The analysis of existing scientific publications has shown that the existing approaches available for the reconstruction of timeline have failed to achieve the major factors of a digital investigation process, such as extraction of events, heterogeneity and huge volume of data, a precisely defined investigation model, analysis capabilities, integrity of information and flexibility. The literature studies have shown as well that the available ontologies of the digital forensics domain do not contain basic and novel terms corresponding to numerous operating systems based devices. These ontologies have been developed to achieve particular goals and objectives. Moreover, these ontologies are not technically verified and validated.
2. Considering the results of the analysis of literature studies, a novel abstraction based approach has been developed for the analysis of the timeline. This approach allows the analysis of the timeline to reduce the complexity of a timeline by splitting the timeline into four relevant levels of the timeline of events and artefacts. Additionally, the analysis of the timeline allows the digital investigator to interpret the underlying information in the timeline easily and effectively. The splitting of the timeline is based on the number of events and artefacts available in the timeline along with their significance or importance and the level of detail (abstraction) of information on the timeline.
3. A new ontology has been developed as well by considering the outcome of literature analysis of the existing ontologies of the digital forensics domain. The developed ontology facilitates the digital investigator in understanding the significances of basic terms of the digital forensics domain and new terminologies corresponding to different operating systems based devices simultaneously. Moreover, the developed ontology is technically evaluated.
4. The developed abstraction based approach was developed in object-oriented programming language, i.e., Java, and implemented on three different operating systems based devices, namely Windows, Android, iOS, during the experimental study of research. The findings of experimental studies show that the proposed approach is capable to reduce the complexity of the timeline generated by the digital forensics tools and addresses the key issues of digital investigation, i.e., the extraction of events and information from timeline, heterogeneity and huge volume of sources of data, clearly defined investigation model, analysis capabilities, the integrity of data and flexibility. Moreover, comparative studies of the developed approach for the timeline analysis and the existing approaches for the reconstruction of timeline based on 21 different factors or features have been performed as well. The findings of comparative studies show that a novel approach is capable of analysing the timeline more effectively and appropriately than the existing approaches. The proposed approach accomplish all (21) features and provides complete information related to all online and offline actions, available user and system

files, applications executed by the user, mail addresses and contact numbers used for the communication or other purpose and much more.

5. The novel ontology is developed by using an ontology editor, i.e., Protégé 5.5.0 Build Beta 9 version, along with various visualization plugins, namely OWLViz, OntoGraf, and VOWL and OWLGred. The developed ontology contains a base class "Digital_Forensic", three sub-classes, i.e., "Artefact_Investigation", "Artefact_Location" and "Artefact_Reference". The "Artefact_Investigation" illustrates the basic terms of the digital forensics domain, "Artefact_Location" sub-class shows the novel abstraction based approach for the timeline analysis, and "Artefact_Reference" presents the new terms corresponding to different operating systems based devices that are encountered during the analysis of the timeline. The ontology as well contains two object properties and 18 data properties. The outcome of the analysis of the timeline shows that there are eleven new terminologies that have been encountered during the analysis of the timeline of Windows, Android and iOS operating systems based devices. Moreover, the developed ontology is as well verified and validated by using the ontology taxonomy evaluation method and ontology content evaluation method, respectively.
6. The outcome of the proposed approach for the timeline analysis is in the form of the textual information. Thus, the results, related to numerous activities performed by a user, are not possible to depict in a single figure or image. For this, the visualization technique is implemented on the outcome of the proposed approach to illustrate the information corresponding to numerous activities graphically in a figure. The outcome of visualization has shown that the user is able to attain and interpret information related to multiple activities carried out by a user on a specific digital device at the same unit of time from the single figure of the outcome precisely.

LITERATURE

1. ADEDAYO, O. M. Big data and digital forensics. *IEEE International Conference on Cybercrime and Computer Forensic (ICCCF), Vancouver, BC, 2016, pp. 1-7.*
2. AHMED, R. & R. V. DHARASKAR. Mobile Forensics: An Introduction from Indian Law Enforcement Perspective. *International Conference on Information Systems, Technology and Management*, Springer, Berlin, Heidelberg, 2009.
3. AKREMI, A., M. F. SRITI, H. SALLAY & M. ROUACHED. Ontology-Based Smart Sound Digital Forensics Analysis for Web Services. *International Journal of Web Services Research*, 2019, vol. 16, pp. 70–92.
4. AL- DHAQM, A. M. R., S. H. OTHMAN, S. A. RAZAK & A. NGADI. Towards Adapting Metamodelling Technique for Database Forensics Investigation Domain. *International Symposium on Biometrics and Security Technologies (ISBAST)*, Kuala Lumpur, 2014, pp. 322-327.
5. ALZAABI, M., A. JONES & T. A. MARTIN. An Ontology-Based Forensic Analysis Tool. *In Proceedings of the Annual ADFSL Conference on Digital Forensics, Security and Law*, Richmond, Virginia, 2013, pp. 10–12.
6. ALZAABI, M., T. A. MARTIN, K. TAHA & A. JONES. The Use of Ontologies in Forensic Analysis of Smartphone Content. *The Journal of Digital Forensics, Security and Law*, 2015, vol. 10, pp. 105–114.
7. AMATO, F., G. COZZOLINO, A. MAZZEO & N. MAZZOCCA. Correlation of Digital evidences in forensic investigation through semantic technologies. *In Proceedings of the 31st International Conference on Advanced Information Networking and Applications Workshops*, Taipei, Taiwan, 2017, pp. 27–29.
8. ANGELINI, M. et al. The goods, the bads and the uglies: Supporting decisions in malware detection through visual analytics, IEEE symposium on visualization for cyber security, Phoenix, AZ, USA, 2017, pp. 1–8.
9. ARSHAD, H., A. JANTAN & O. I. ABIODUN. Digital Forensics: Review of Issues in Scientific Validation of Digital Evidence, *Journal of Information Processing Systems*, 2018, vol. 14, nr. 2, pp. 346-376.
10. BANG, J., B. YOO, J. KIM & S. LEE. Analysis of time information for digital investigation. *In Proceedings of the Fifth International Joint Conference on INC, IMS and IDC*, Seoul, Korea, 2009, pp. 1858–1864.
11. BASSETT, R., L. BASS & P. O'BRIEN. Computer forensics: an essential ingredient for cyber security. *Journal of Information Science and The Technology*, 2006, vol. 3, nr. 1, pp. 22-32.
12. BEEBE, N. Digital Forensic Research: The Good, the Bad and the Unaddressed. *International Conference on Digital Forensics, Orlando*, Florida, USA, 2009, pp. 17-36.
13. BEEBE, N. L. & J. G. CLARK. A hierarchical, objectives-based framework for the digital investigations process. *Digital Investigation*, 2005, vol. 2, nr. 2, pp. 147-167.
14. BEEBE, N. L., J. G. CLARK, G. B. DIETRICH, M. S. KO. & D. KO. Post-retrieval search hit clustering to improve information retrieval effectiveness: Two digital forensics case studies, *Decision Support Systems*, 2011, vol.51, nr. 4, pp. 732-744.
15. BERGMAN, M. Conceptual and Practical Distinctions in the Attributes Ontology [online]. 2015 [viewed 15 September 2020]. Available from: https://www.mkbergman.com/1842/conceptual-and-practical-distinctions-in-the-attributes-ontology.

16. BHANDARI, S. & V. JUSAS. An ontology based on the timeline of Log2timeline and Psort using abstraction approach in digital forensics. *Symmetry*, 2020, vol. 12, nr. 4, 642.
17. BHANDARI, S. & V. JUSAS. An abstraction based approach for reconstruction of timeline in digital forensics. *Symmetry*, 2020, vol. 12, 104.
18. BHANDARI, S. &, V. JUSAS. An audit: Digital forensic research. *International Journal of Advances in Electronics and Computer Science*, 2019, vol. 6, pp. 71-75.
19. BRADY, O., R. OVERILL & J. KEPPENS. Addressing the increasing volume and variety of digital evidence using an ontology. *In Proceedings of the 2014 IEEE Joint Intelligence and Security Informatics Conference*, Hague, The Netherlands, 2014, pp. 176–183.
20. BRADY, O., R. OVERILL & J. KEPPENS. DESO: Addressing volume and variety in large scale criminal cases. *Digital Investigation*, 2015, vol. 15, pp. 72–82.
21. BRINSON, A., A. ROBINSON &. M. ROGER. A Cyber Forensics Ontology: Creating a New Approach to Studying Cyber Forensics. *Digital Investigation*, 2006, vol. 3, pp. 37–43.
22. CARRIER, B. & E. H. SPAFFORD. Getting Physical with the Digital Investigation Process. *International Journal of Digital Evidence*, 2003, vol. 2, nr. 2.
23. CARRIER, B. D. & E. H. SPAFFORD. An Event-Based Digital Forensic Investigation Framework. *The Digital Forensic Research Conference DFRWS*, Baltimore, MD, USA, 2004.
24. CARRIER, B. Defining Digital Forensic Examination and Analysis Tools. *The Digital Forensic Research Conference DFRWS*, Syracuse, NY, USA, 2002.
25. CARVEY, H. *Windows Forensic Analysis DVD Toolkit 2E*. Burlington, MA, USA: Syngress, 2009.
26. CASE, A., A. CRISTINA, L. MARZIALE, G. G. RICHARD & V. ROUSSEV. FACE: automated digital evidence discovery and correlation. *Digital Investigation*, 2008, vol. 5, pp. S65-S75.
27. CASEY, E. *Handbook of Digital Forensics and Investigation*. Orlando, FL, United States: Academic Press, Inc.6277 Sea Harbor Drive, 2009.
28. CASEY, E. State of the field: growth, growth, growth. *Digital Investigation*, 2004, vol. 1, pp. 241-242.
29. CASEY, E., M. FERRARO &, L. NGUYEN. Investigation Delayed Is Justice Denied: Proposals for Expediting Forensic Examinations of Digital Evidence. *Journal of Forensic Sciences*, 2009, vol. 54, nr. 6, pp. 1353-1364.
30. CAVIGLIONE, L., S. WENDZEL &, W. MAZURCZYK. The Future of Digital Forensics: Challenges and the Road Ahead. *IEEE Security and Privacy Magazine*, 2017, vol. 15, nr. 6, pp. 12-17.
31. CHABOT, Y., A. BERTAUX, C. NICOLLE & M. T. KECHADI. A complete formalized knowledge representation model for advanced digital forensics timeline analysis. *Digital Investigation*, 2014, vol. 11, pp. 95–105.
32. CHABOT, Y., A. BERTAUX, C. NICOLLE & M. T. KECHADI. An ontology-based approach for the reconstruction and analysis of digital incidents timelines. *Digital Investigation*, 2015, vol. 15, pp. 83–100.
33. CHANDRAWANSHI, R. & H. GUPTA. A Survey : Server timeline analysis for web forensics. *International Journal of Scientific Research Engineering & Technology*, 2013, vol. 1, nr. 12, pp. 017-021.

34. CHAVAN, S. K. & S. M. NIRKHI. Visualization techniques for digital forensics : a survey, International journal of advanced computer research, 2012, vol. 2, nr. 6, pp. 74-78.
35. CHEN, K., A. J. CLARK, O. D. VEL & G. MOHAY. ECF – Event Correlation for Forensics. *1st Australian Computer Network & Information Forensics Conference*, Perth, West Australia, 2003, pp. 1-10.
36. CHEN, M., et al. Data, Information, and Knowledge in Visualization, *IEEE Computer Graphics and Applications*, 2009, vol. 29 nr. 1, pp. 12-19.
37. CHO, G. S. A computer forensic method for detecting timestamp forgery in NTFS. *Computers & Security*, 2013, vol. 34, pp. 36-46.
38. CHOPADE, R. & V.K. PACHGHARE. Ten years of critical review on database forensics research. *Digital Investigation*, 2019, vol. 29, pp. 180-197.
39. COHEN, M. I. Pyflag – an advanced network forensic framework. *Digital Investigation*, 2008, vol. 5, pp. 112-120.
40. ĆOSIĆ, J. & Z. ĆOSIĆ. The Necessity of Developing a Digital Evidence Ontology. *In Proceedings of the 23rd Central European Conference on Information and Intelligent Systems*, Varazdin, Croatia, 2012, pp. 325–330
41. ĆOSIĆ, J., Z. ĆOSIĆ & M. BAČA. An Ontological Approach to Study and Manage Digital Chain of Custody of Digital Evidence. *Journal of Information and Organizational Sciences*, 2011, vol. 35, pp. 1–13.
42. CURRAN, K., A. ROBINSON & S. PEACOCKE. Cassidy. Mobile Phone Forensic Analysis. *International Journal of Digital Crime and Forensics*, 2010, vol. 2, nr. 2.
43. DEBINSKI, M., F. BREITINGER & P. MOHAN. Timeline2GUI: A Log2timeline CSV parser and training scenarios. *Digital Investigation*, 2019, vol. 28, pp. 34–43.
44. ELHADAD, M. K., K. BADRAN & G. I. SALAMA. A novel approach for ontology-based dimensionality reduction for web text document classification. *IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, Wuhan, China, 2017, pp. 373-378.
45. *Email Market, 2020-2024* [online]. 2020 [viewed 04 November 2020]. Available from: https://www.radicati.com/wp/wp-content/uploads/2020/04/Email-Market-2020-2024-Executive-Summary.pdf.
46. *EnCase tool* [online]. 2020 [viewed 25 June 201]. Available from: https://www.guidancesoftware.com/encase-forensic.
47. ESPOSITO, S. &, G. PETERSON. Creating Super Timelines in Windows Investigations. *In Proceedings of the 9th International Conference on Digital Forensics*, Orlando, FL, USA, 2013, pp. 135–144.
48. GANDHI, P. & J. PRUTHI. Data Visualization Techniques: Traditional Data to Big Data, In: Data Visualization, Springer, Singapore, 2020.
49. GARFINKEL, S. L. Digital forensics research: The next 10 years. *Digital Investigation*, 2010, vol. 7, pp. S64-S73.
50. GARFINKEL, S. L. Forensic feature extraction and cross-drive analysis. *Digital Investigation*, 2006, vol. 3, pp. 71-81.
51. GHAZINOUR. K., D. M. VAKHARIA, K. C. KANNAJI & R. SATYAKUMAR. A study on digital forensic tools. *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, India, 2017, pp. 3136-3142.

52. GIOVA, G. Improving Chain of Custody in Forensic Investigation of Electronic Digital Systems. *International Journal of Computer Science and Network Security*, 2011, vol. 11, pp. 1-9.
53. GÓMEZ-PÉREZ, A. Knowledge sharing and reuse. In: *Handbook of Applied Expert Systems*. Boca Raton, United States of America: CRC Press, 1998, vol. 10, pp. 1–36.
54. GRIGALIUNAS, S. & A. ZYKAS. *Cyber attacker profiling based on social groups habits evidence* [online]. 2016 [viewed 25 May 2021]. Available from: http://kitm.lt/wp-content/uploads/2016/09/Cyber-Attacker-profiling-based-on-social-groups-habits-evidences.pdf.
55. GRIGALIUNAS, S. & J. TOLDINAS. Digital evidence investigation using habits attribution. *The 4th International Virtual Research Conference in Technical Disciplines*, 2016, pp. 30-35.
56. GRIGALIUNAS, S., J. TOLDINAS & A. VENCKAUSKAS. An ontology-based transformation model for the digital forensics domain. *Elektronika Ir Elektrotechnika*, 2017, vol. 23, nr. 3, pp. 78-82.
57. GRUBER, T. Towards Principles for the Design of Ontologies Used for Knowledge Sharing?. *International Journal of Human-Computer Studies*, 1995, vol. 43, pp. 907–928.
58. GRUBER, T.R. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 1993, vol. 5, pp. 199–220.
59. GUÐJÓNSSON, K. *Mastering the Super Timeline Who Am I ?* [online]. 2010 [viewed 22 May 2020]. Available from: https://digital-forensics.sans.org/summit-archives/2010/eu-digital-forensics-incident-response-summit-kristinn gudjonsson-mastering-the-super-timeline.pdf.
60. GUO, H., B. JIN & D. HUANG. Research and review on computer forensics, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 2010, vol. 56, pp. 224–233.
61. HADZIC, M., P. WONGTHONGTHAM, T. DILLON & E. CHANG. Introduction to Ontology. In : Ontology-Based Multi-Agent Systems. *Studies in Computational Intelligence*. Berlin/Heidelberg, Germany: Springer, 2009, vol. 219.
62. HARGREAVES, C. & J. PATTERSON. An automated timeline reconstruction approach for digital forensic investigations. *Digital Investigation*, 2012, vol. 9, pp. S69–S79.
63. HARRELL, C. *What's a Timeline?* [online]. 2011 [viewed 28 May 2019]. Available from: http://journeyintoir.blogspot.com/2011/09/whats-timeline.html.
64. HARRILL, D.C. & R. P. MISLAN. A Small Scale Digital Device Forensics ontology. *Small Scale Digital Device Forensics Journal*, 2007, vol.1, pp. 1–7.
65. HARRISON, L. & A. LU. The future of security visualization: Lessons from network visualization, IEEE Network , 2012, vol. 26, nr. 6, pp.6–11.
66. HEVNER, A. R., S. T. MARCH, J. PARK & S. RAM. Design Science in Information Systems Research. *MIS Quarterly*, 2004, vol. 28, nr. 1, pp.75-105.
67. HOSS, A. M. & D. L. CARVER. Weaving Ontologies to Support Digital Forensic Analysis. *IEEE International Conference on Intelligence and Security Informatics*, Dallas, TX, USA, 2009, pp. 203-205.
68. HUANG, J., A. YASINSAC & P. J. HAYES. Knowledge Sharing and Reuse in Digital Forensics. *In Proceedings of the 2010 Fifth IEEE International Workshop on Systematic Approaches to Digital Forensic Engineering*, Oakland, CA, USA, 2010, pp. 1–6.

69. IEONG, R. S. C. FORZA – Digital forensics investigation framework that incorporate legal issues. *Digital Investigation*, 2006, vol. 3, pp. S29-S36.
70. INGLOT, B. &, L. LIU. Enhanced Timeline Analysis for Digital Forensic Investigations. *Information Security Journal: A Global Perspective*, 2014, vol. 23, pp. 32–44.
71. JAMES, J.I. & P. GLADYSHEV. Automated inference of past action instances in digital investigations. *International Journal of Information Security*, 2015, vol. 14, pp. 249–261.
72. JANG, Y. & J. KWAK. Digital forensics investigation methodology applicable for social network services. *Multimedia Tools and Applications*, 2015, vol. 74, pp. 5029-5040.
73. JANSEN, W., & AYERS, R. *Guidelines on PDA Forensics, Recommendations of the National Institute of Standards and Technology (NIST)* [online]. 2004 [viewed 04 November 2020] Available from: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-72.pdf.
74. JENSEN, W. B. Classification, symmetry and the periodic table. *Computers & Mathematics with Applications*, 1986, vol. 12, pp. 487-510.
75. KAHVEDŽIC, D. & T. KECHADI. DIALOG: A framework for modeling, analysis and reuse of digital forensic knowledge. *Digital Investigation*, 2009, vol. 6, pp. S23–S33.
76. KALBER, S., A. DEWALD & F. C. Freiling. Forensic application-fingerprinting based on file system metadata. *In Proceedings of the Seventh International Conference on IT Security Incident Management and IT Forensics*, Nuremberg, Germany, 2013, pp. 98–112.
77. KALEMI, E. & S. Y. YAYILGAN. Ontologies for Social Media Digital Evidence. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 2016, vol. 10, pp. 324–329.
78. KARIE, N. M. & H. S. VENTER. Taxonomy of Challenges for Digital Forensics. *Journal of Forensic Sciences*, 2015, vol 60, nr. 4, pp. 885-893.
79. KARIE, N. M. & H. S. VENTER. Toward a General Ontology for Digital Forensic Discipline. *Journal of Forensic Sciences*, 2014, vol. 59, pp. 1231-1241.
80. KARIE, N.M. & V. R. KEBANDE. Building Ontologies for Digital Forensic Terminologies. *International Journal of Cyber-Security and Digital Forensics*, 2016, vol. 5, pp. 75–82.
81. KEIM, D. A. Information Visualization and Visual Data mining. IEEE Transactions on visualization and computer graphics, 2002, vol. 8, no. 1, pp. 1-8.
82. KEIM, D. A., F. MANSMANN, J. SCHNEIDEWIND & H. ZIEGLER. Challenges in Visual Data Analysis. *Tenth International Conference on Information Visualisation (IV'06)*, London, England, UK, 2006, pp. 9-16.
83. KHAN, M. N. A., C. R. CHATWIN & R. C. D. YOUNG. A framework for post-event timeline reconstruction using neural networks. *Digital Investigation*, 2007, vol. 4, pp. 146-157.
84. KUMARI, N. & A. K. MOHAPATRA. An insight into digital forensics branches and tools. *International Conference on Computational Techniques in Information and Communication Technologies*, New Delhi, India, 2016, pp. 243–250.
85. LUTHFI, A. The Use of Ontology Framework for Automation Digital Forensics Investigation. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 2014, vol. 8, pp. 454–456.
86. MANOVICH, L. What is Visualization?. *Visual Studies*, 2011, vol. 26, nr.1, pp. 36-49.

87. MARZIALE, L., G. G. RICHARD & V. ROUSSEV. Massive threading: Using GPUs to increase the performance of digital forensics tools. *Digital Investigation*, 2007, vol. 4, pp. 73-81.
88. MISTRY, N. & A. PATEL. An analyzing of different techniques and tools to recover data from volatile memory. *International journal for scientific research & development*, 2013, vol. 1, nr. 2, pp. 227-233.
89. MOHAY, G. Technical Challenges and Directions for Digital Forensics. *Proceedings of the First International Workshop on Systematic Approaches to Digital Forensic Engineering*, Taipei, Taiwan, 2005.
90. MUKKAMALA, S. & A. H. SUNG. Identifying Significant Features for Network Forensic Analysis Using Artificial Intelligent Techniques. *International Journal of Digital Evidence*, 2003, vol. 1, nr. 4.
91. NAJLA, S., J. WASSIM & G. FAIEZ. Extension of Protégé to support evolution of ontology. *First International Conference on Advances in Databases, Knowledge, and Data Applications*, Gosier, Guadeloupe/France, 2009, pp. 149-154.
92. NALAWADE, A., S. BHARNE & V. MANE. Forensic Analysis and Evidence Collection for Web Browser Activity. *International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, Pune, India, 2016, pp. 518-522.
93. NANCE, K., B. HAY & M. BISHOP. Digital Forensics: Defining a Research Agenda. *42nd Hawaii International Conference on System Sciences*, Big Island, HI, USA, 2009.
94. National Institute of Justice. Electronic Crime Scene Investigation: A Guide for First Responders, Second Edition, *NIJ Research Report*, 2008.
95. NIRKHI, S. Potential use of artificial neural network in data mining. International conference on computer and automation engineering (ICCAE), Singapore, 2010, pp 339- 343.
96. NOY, N.F. & D. L. MCGUINNESS. *Ontology Development 101: A Guide to Creating Your First Ontology* [online]. 2001 [viewed 25 January 2020] Available from: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf.
97. OLAJIDE, F., N. SAVAGE, D. NDZI & H. AL-SINANI. *Forensic Live Response and Event Reconstruction Methods in Linux Systems* [online]. 2009 [viewed 25 May 2020] Available from: http://www.cms.livjm.ac.uk/pgnet2009/Proceedings/Papers/2009001.pdf.
98. OLSSON, J. & M. BOLDT. Computer forensic timeline visualization tool. *Digital Investigation*, 2009, vol. 6, pp. S78-S87.
99. Ontology in Computer Science. In: Semantic Web: Concepts, Technologies and Applications. *NASA Monographs in Systems and Software Engineering*. London: Springer, London, 2007, [viewed 28 October 2020]. Available from: https://doi.org/10.1007/978-1-84628-710-7_2.
100. OSBORNE, G. & B. TURNBALL. Enhancing computer forensics investigation through visualisation and data exploitation. International conference on availability, reliability, and security, Fukuoka, Japan, 2009, pp. 1012-1017.
101. OSBORNE, G., H. THINYANE & J. SLAY. Visualizing Information in Digital Forensics. In: Peterson, G., S. Shenoi. Advances in Digital Forensics VIII. Digital Forensics 2012. *IFIP Advances in Information and Communication Technology*. Berlin, Heidelberg: Springer, 2012, vol.383.
102. PALMER, G. A Road Map for Digital Forensic Research. *Report from DFRW First Digital Forensic Research Workshop, Utica*, New York, 2001, pp. 27–30.

103. PANDEY, A. K. et al. Current Challenges of Digital Forensics in Cyber Security. In: Critical Concepts, Standards, and Techniques in Cyber Forensics, Project: Software Security, Lab: Information Technology Laboratory, 2020.
104. PARK, H., S. CHO & H. C. KWON. Cyber Forensics Ontology for Cyber Criminal Investigation. *Proceedings of the International Conference on Forensics in Telecommunications, Information, and Multimedia*, Adelaide, Australia, 2009, pp. 160–165.
105. POISEL, R. & S. TJOA. Forensics Investigations of Multimedia Data: A Review of the State-of-the-Art, *Sixth International Conference on IT Security Incident Management and IT Forensics*, Stuttgart, 2011, pp. 48-61.
106. POLLITT, M. A History of Digital Forensics. *IFIP International Conference on Digital Forensics*, Hong Kong, China, 2010, pp. 3-15.
107. *Practice advice on core investigative doctrine* [online]. (2005) [viewed 04 November 2020] Available from: http://library.college.police.uk/docs/acpo/Core-Investigative-Doctrine.pdf.
108. PRASAD, M.A.R. & Y.N. SATISH. Reconstruction of Events in Digital Forensics. *International Journal of Engineering Trends and Technology*, 2013, vol. 4, pp. 3460–3467.
109. PURCHASE, H. C., N. ANDRIENKO, T. J. JANKUN-KELLY & M. WARD. Theoretical Foundations of Information Visualization. In: Kerren A., J. T. Stasko, JD Fekete & C. North. *Information Visualization. Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, 2008, vol 4950.
110. QUICK, D. & K. K. R. CHOO. Impacts of increasing volume of digital forensic data: A survey and future research challenges. *Digital Investigation*, 2014, vol. 11, nr. 4, pp. 273–294.
111. RAGHAVAN, S. Digital forensic research: current state of the art. *CSI Transactions on ICT*, 2012, vol.1, pp. 91-114
112. RAHIM, N., A. W. A. WAHAB, M. Y. I. IDRIS & M. L. M. KIAH. Digital Forensics: An Overview of the Current Trends. *International Journal of Cryptology Research*, 2014, vol. 4, nr. 2.
113. REITH, M., C. CARR & G. GUNSCH. An Examination of Digital Forensic Models. *International Journal of Digital Evidence*, 2002, vol.1, nr. 3.
114. RUBIO, E. Probabilistic self-organizing maps for continuous data. IEEE transactions on neural networks, 2010, pp. 1543 – 1554.
115. RUSSELL, S.J. & P. NORVIG. *Artificial Intelligence: A Modern Approach*. Upper Saddle River, New Jersey, United States of America: Prentice Hall, 2016.
116. RUTGERS, S. T., & M. A. VASARHELYI. Cluster analysis for anomaly detection in accounting data: an audit approach. The international journal of digital accounting research, 2011, pp. 69-84.
117. SAAD, S. & I. TRAORE. Method ontology for intelligent network forensics analysis. *In Proceedings of the 8th International Conference on Privacy, Security and Trust*, Ottawa, ON, Canada, 2010, pp. 7–14.
118. SALUNKHE, P., S. BHARNE & P. PADIYA. Data Analysis of File Forensic Investigation. *International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Paralakhemundi, India, 2016, pp. 372-375.
119. SHARMA, B. K., M. A. JOSEPH, B. JACOB & B. MIRANDA. Emerging trends in Digital Forensic and Cyber security- An Overview. *Sixth HCT Information Technology Trends*, Ras Al Khaimah, United Arab Emirates, 2019, pp. 309-313.

120. SHIN, Y. New Digital Forensics Investigation Procedure Model. *Fourth International Conference on Networked Computing and Advanced Information Management*, Gyeongju, South Korea, 2008, pp. 528-531.
121. SHOSHA, A. F., L. TOBIN & P. GLADYSHEV. Digital Forensic Reconstruction of a Program Actions. *IEEE Security and Privacy Workshops*, San Francisco, CA, USA, 2013, pp. 119-122.
122. SHRIVASTAVA, A. K., N. PAYAL, A. RASTOGI & A. TIWARI. Digital Forensic Investigation Development Model. *5th International Conference on Computational Intelligence and Communication Networks*, Mathura, India, 2013, pp. 532-535.
123. SITOMPUL, O.S., A. HANDOKO & R. F. RAHMAT. File Reconstruction in Digital Forensic. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, 2018, vol. 16, pp. 776–794.
124. SOLTANI, S., S. A. H. SENO & H. S. YAZDI. Event reconstruction using temporal pattern of file system modification. *IET Information Security*, 2019, vol. 13, pp. 201–212.
125. SPIEKERMANN, D., J. KELLER & T. EGGENDORFER. Network forensic investigation in OpenFlow networks with ForCon. *Digital Investigation*, 2017, vol. 20, pp. S66-S74.
126. SUBEKTININGSIH, S., Y. PRAYUDI & I. RIADI. Digital Forensics Workflow as A Mapping Model for People, Evidence, and Process in Digital Investigation. *International Journal of Cyber-Security and Digital Forensics*, 2018, vol. 7, nr. 3, pp. 294-304.
127. TANNER, A. L. & D. A. DAMPIER. An Approach for Managing Knowledge in Digital Forensic Examinations. *International Journal of Computer Science and Security*, 2010, vol. 4, nr. 5, pp. 451-465.
128. THOMAS, D.S. & K.A. FORCHT. Legal methods of using computer forensics techniques for computer crime analysis and investigation. *Issues in Information Systems*, 2004, vol. 5, nr. 2, pp. 692–698.
129. TURNBULL, B. & S. RANDHAWA. Automated event and social network extraction from digital evidence sources with ontological mapping. *Digital Investigation*, 2015, vol. 13, pp. 94-106.
130. *Vocabularies and Ontologies* [online]. 2015 [viewed 05 November 2020] Available from: https://ceweb.br/guias/web-semantica/en/capitulo-6/.
131. WANG, Y.Q. & M. QI. Computer forensics in communication networks. International communication conference on wireless mobile and computing, Shanghai, 2011, pp. 379-383, 2011.
132. WHITCOMB, C. M. An Historical Perspective of Digital Evidence: A Forensic Scientist's View. *International Journal of Digital Evidence*, 2002, vol. 1, nr. 1.
133. WIMMER, H., L. CHEN &, T. NAROCK. Ontologies and the Semantic Web for Digital Investigation Tool Selection. *The Journal of Digital Forensics, Security and Law*, 2018, vol. 13, pp. 21–46.
134. XIE, G. et al. Resurf: reconstructing web-surfing activity from network traffic. *IFIP networking Conference*, Brooklyn, NY, 2013, pp. 1-9.
135. YATES, M. Practical Investigations of Digital Forensics Tools for Mobile Devices. *Proceedings of the Information Security Curriculum Development Annual Conference*, Kennesaw, GA, USA, 2010, pp. 156-162.

136. ZHANG, H., L. CHEN & Q. LIU. Digital Forensic Analysis of Instant Messaging Applications on Android Smartphones. *International Conference on Computing, Networking and Communications (ICNC)*, Maui, HI, USA, 2018, pp. 647-651.
137. ZHAO, K., B. LIU, T. M. TIRPAK & A. SCHALLER. Detecting patterns of change using enhanced parallel coordinates visualization. IEEE International Conference on Data Mining, Melbourne, FL, USA, 2003, pp. 747-750.
138. ZHAO, L. & J. O. COPLIEN. Understanding symmetry in object-oriented languages. *Journal of Object Technology*, 2003, vol. 2, nr. 5, pp. 123-134.
139. ZHAO, Y. et al. A Survey on Network Security Data Visualization, Journal of Computer-Aided Design and Computer Graphics, 2014, vol. 26, pp. 687–697.

LIST OF PUBLICATIONS

Articles in *Web of science* database publications having citation index:

1. BHANDARI, S. & V. JUSAS. An abstraction based approach for reconstruction of timeline in digital forensics. *Symmetry*, 2020, vol. 12, nr. 1.
2. BHANDARI, S. & V. JUSAS. An ontology based on the timeline of Log2timeline and Psort using abstraction approach in digital forensics. *Symmetry*, 2020, vol. 12, nr. 4.

Articles in Web of Science database without citation index:

1. BHANDARI, S. & V. JUSAS. The phases based approach for regeneration of timeline in digital forensics. *2020 International conference on innovations in intelligent systems and applications (INISTA)*, Novi Sad, Serbia, 2020, pp. 1-6.
2. BHANDARI, S. & A. KULIKAJEVAS. Ontology based image recognition: a review. *IVUS 2018: proceedings of the international conference on information technologies*, Kaunas, Lithuania, 2018, vol. 2145, pp. 13-18.
3. BHANDARI, S. & V. JUSAS. An audit: digital forensics research. *Proceedings of IASTEM international conference*, Krakow, Poland, 2019, pp. 7-11.
4. BHANDARI, S. & V. JUSAS. Enhancement of timeline analysis for digital forensics. *Proceedings of IASTEM international conference*, Krakow, Poland, 2019, pp. 1-6.
5. BHANDARI, S. & V. JUSAS. Examination and classification of data in digital forensics. *10th international workshop on data analysis methods for software systems*, Druskininkai, Lithuania, 2018, pp. 11.

AWARDS

The author of the dissertation participated in KTU competition and was elected as the most active PhD student in 2020.

CURRICULUM VITAE

Personal details:
Full name: Sandeepak Bhandari
Date of birth: 28th March 1990
E-mail: sandeepak525@gmail.com

Education:
2010–2013 Bachelor of Technology in Information Technology at I.K. Gujral Punjab Technical University.
2013–2016 Master of Technology in Computer Science and Engineering at I.K. Gujral Punjab Technical University.
2017–2022 Doctoral studies in Informatics Engineering at Kaunas University of Technology.

SANTRAUKA

1. ĮŽANGA

Teisinis skaitmeninis įrodymų tyrimas – tai skaitmeninių įrodymų, aptinkamų skaitmeniniuose įrenginiuose, identifikavimo, rinkimo, išsaugojimo, analizės ir pateikimo teisminiam nagrinėjimui procesas. Tam, kad būtų galima surinkti skaitmeninius įrodymus tiriant skaitmeninius nusikaltimus, reikia atlikti laiko juostų rekonstrukciją (laiko analizę). Be to, laiko analizė padeda nustatyti daugybę veiksmų, kuriuos vartotojas atliko ar atlieka konkrečioje sistemoje. Norint atlikti laiko analizę ir interpretuoti gautą informaciją, kad būtų galima surinkti skaitmeninius įrodymus, tenka išanalizuoti daugybę įvykių, kurių gausą lemia sparčiai besiplečiantis žiniatinklis, tarpusavyje susieti įrenginiai, didžiulis duomenų kiekis, duomenų įvairovė, naujoviškos technologijos ir virtinė kitų veiksnių. Literatūros tyrimai rodo, kad yra sukurta daug laiko analizės metodų, padedančių skaitmeninės srities specialistams atlikti laiko analizę ir interpretuoti informaciją bei rinkti skaitmeninius įrodymus, vis dėlto nė vienas iš tokių metodų nepadeda išspręsti iššūkių, su kuriais susiduria nusikalstamos veikos kibernetinėje erdvėje tyrėjai (kriminalistai), siekiantys kompetentingai ištirti įrodymus ir išanalizuoti laiko juostas.

Atlikdami laiko analizę, tyrėjai susiduria su įvairiais naujais terminais, kurių atsiradimą lemia technologijų naujovės, duomenų nevienalytiškumas ir daugelis kitų veiksnių. Be to, teisinio skaitmeninių įrodymų nagrinėjimo srityje naudojamos priemonės lėmė nestruktūruotas laiko analizes, pagrįstas duomenų šaltinių gausa. Tokiais atvejais laikotarpis, reikalingas potencialaus skaitmeninio incidento priežasčiai surasti ir interpretuoti, gali priklausyti nuo to, kaip sudėtinga yra suprasti naujų terminų reikšmę. Siekiant išspręsti šias problemas ir padėti tyrėjams tirti kibernetinėje erdvėje įvykdomus nusikaltimus, šioje tyrimų srityje yra sukurti du metodai: pirma, abstrakcija pagrįstas laiko analizės metodas ir, antra, ontologija, skirta atsiradusiems terminams (terminijoms), su kuriais susiduriama atliekant tokias analizes, naujai apibrėžti.

1.1. Motyvacija

Rengiant šią disertaciją, siekta sukurti naują abstrakcija pagrįstą laiko analizės metodą. Pirmoji tokio sprendimo priežastis yra ta, kad skirtingi mokslininkų ar autorių sukurti laiko juostų rekonstrukcijos metodai nesprendžia pagrindinių skaitmeninio nusikaltimo tyrimo klausimų, tokių kaip automatinis įvykių ir informacijos identifikavimas laiko juostoje, nevienalytiškumas ir milžiniški duomenų šaltinių kiekiai, aiškiai apibrėžtas tyrimo modelis, analizės galimybės ir duomenų vientisumas. Kita priežastis yra ta, kad literatūros tyrimas parodė, jog įrenginių su keliomis operacinėmis sistemomis atžvilgiu nėra prieinamų laiko juostos rekonstravimo metodų. Trečia priežastis – joks žinomas metodas nenaudoja abstrakcijos sąvokos laiko analizei atlikti..

Suformuluotame metode laiko analizė yra labai svarbi siekiant sumažinti laiko juostos sudėtingumą suskaidant ją į skirtingus ir atitinkamus laiko juostos įvykių ir artefaktų lygius. Remiantis siūlomu metodu, laiko juosta padalijama į keturis atitinkamus įvykių ir artefaktų laiko juostos lygius, tai *Įvykiai: aukštas lygis* (nauji įrašai ir naršymas žiniatinklyje), *Įvykiai: žemas lygis* (naršymas žiniatinklyje, modifikavimo veiksmai), Artefaktų lokacija: aukštas lygis (apima visus programos failus) ir *Artefaktų lokacija: žemas lygis*. Pagrindinė laiko juostos suskirstymo į keturis abstrakcijos lygius paskirtis yra pateikti skirtingų rūšių informaciją, o kiekvienam lygiui turėtų būti nurodyta skirtinga struktūra kartu su skirtingais informacijos detalumo lygiais, siekiant sumažinti laiko juostos sudėtingumą, praleidžiant nepageidaujamas detales, užtikrinant laiko juostos teisingumą (tikslumą) ir pateikiant tik

informaciją, padedančią atpažinti ir suprasti konkrečius vartotojų atliekamus veiksmus, analizuojant skirtingus šaltinius ir laukus.

Disertacijoje taip pat pristatyta ir techniškai įvertinta nauja ontologija, pagrįsta abstrakciniu laiko analizės metodu. Pirmoji naujosios ontologijos suformavimo priežastis yra ta, kad, atlikus literatūros tyrimą, neaptikta ontologijos, kurią sudarytų svarbiausi skaitmeninės kriminalistikos srities terminai bei naujai sukurti terminai, vartojami skirtingiems įrenginiams su skirtingomis operacinėmis sistemomis. Antroji priežastis yra ta, kad jau suformuluotos ontologijos nėra išanalizuotos techniniu atžvilgiu ir patvirtintos. Sukurtąją ontologiją sudaro svarbiausia ir nauja terminija, atitinkanti operacinių sistemų *Windows, Android* ir *iOS* pagrindu veikiančius įrenginius. Be to, ši ontologija yra išsami ir gali būti lengvai plečiama.

1.2. Tyrimo objektas

Abstrakcijos koncepcija paremto laiko analizės metodo ir ontologijos kūrimas.

1.3. Tikslai ir paskirtis

Papildyti skaitmeninėje kriminalistikoje naudojamą laiko analizę nauju metodu, skirtu laiko juostoms analizuoti abstrakcijos koncepcijos pagrindu.

Disertacijos tikslai:

1. Išanalizuoti literatūrą, susijusią su pagrindiniais terminais, vartojamais skaitmeninės kriminalistikos srityje, ontologija ir vizualizacija, taip pat įvairius esamus laiko juostų rekonstrukcijos metodus ir ontologijas teisinio skaitmeninių įrodymų nagrinėjimo srityje.
2. Sukurti naują abstrakcija pagrįstą laiko analizės metodą, apibrėžiant keturis laiko juostų ir naujosios ontologijos abstrakcijos lygius, taip pat svarbiausius ir naujai aptinkamus terminus (terminijas).
3. Atlikti virtinę eksperimentų ir įvertinti siūlomo metodo ir ontologijos tyrimų rezultatus.

1.4. Mokslinio tyrimo metodika

Tyrimas, atliktas rengiant šią daktaro disertaciją, buvo vykdomas taikant dizaino mokslo tyrimo metodiką. Paprastai dizaino tyrimo metodika naudojama kuriant naują artefaktą ar tobulinant esamus artefaktus, pvz., algoritmus, žmogaus ir kompiuterio sąsajas, dizaino metodikas ir t. t. [66]. Šiame tyrime tai yra laiko analizės metodas ir ontologija, naudotini teisinio skaitmeninių įrodymų nagrinėjimo srityje.

1.5. Disertacijoje ginami teiginiai

Disertacijoje ginami teiginiai yra šie:

1. Naujas abstrakcija pagrįstas metodas leidžia analizuoti laiko juostą, sugeneruotą teisinio skaitmeninių įrodymų nagrinėjimo įrankių, o būtent – *Log2timeline* ir *psort*, padalijant laiko juostą į keturis atitinkamus įvykių ir artefaktų laiko juostos lygius.
2. Naujoji ontologija, paremta siūlomu abstrakcija pagrįstu laiko analizės metodu, leidžia skaitmeninės srities specialistams arba tyrėjams interpretuoti pagrindinius ir naujai aptinkamus teisinio skaitmeninių įrodymų nagrinėjimo srities terminus (terminijas) ir jų reikšmę.

1.6. Disertacijos mokslinis naujumas

1. Jokiame iki šiol publikuotame moksliniame darbe nerasta įrodymų, kad laiko analizės tikslais yra naudota abstrakcijos koncepcija, o sukurta ontologija susideda iš naujų terminų, atitinkančių kalbą, vartojamą įrenginiuose su operacinėmis sistemomis *Windows, Android* ir *iOS*.
2. Naujas abstrakcija pagrįstas metodas mažina laiko juostos sudėtingumą, rekonstruodamas ją keturiais lygiais: aukšto lygio įvykiai, žemo lygio įvykiai, aukšto lygio artefaktai ir žemo lygio artefaktai. Šis metodas taikomas įrenginiuose su operacinėmis sistemomis *Windows, Android* ir *iOS*.
3. Laiko juostos lygiai yra išdėstyti mažėjančios abstrakcijos tvarka, ir kiekvienas lygis suteikia papildomos informacijos ir detalių, praleidžiant nereikšmingas detales ir užtikrinant laiko analizės teisingumą (tikslumą).
4. Sukurtoji ontologija susideda iš naujai aptinkamų terminų, atitinkančių kalbą, vartojamą įrenginiuose su operacinėmis sistemomis *Windows, Android* ir *iOS*. Be to, sukurtoji ontologija gali būti toliau plėtojama; ji yra išsami ir glausta.

1.7. Praktinis panaudojimas

Sprendimas, sukurtas atliekant mokslinį tyrimą, leidžia skaitmeninių įrodymų nagrinėjimo specialistui ar vartotojui efektyviai interpretuoti įvairias vartotojo skaitmeniniuose įrenginiuose vykdomas veiklas. Be to, sukurtasis metodas gali būti įgyvendinamas įvairiuose įrenginiuose su operacinėmis sistemomis ir leidžia gauti laiko analizės informaciją. Naujasis metodas įgyvendinamas ir patvirtinamas naudojant realius duomenis, t. y. įrenginių su operacinėmis sistemomis *Windows, Android* ir *iOS* duomenis. Rezultatai rodo, kad sukurtasis metodas gali būti pritaikytas ir įrenginiams su kitomis operacinėmis sistemomis, jis leidžia efektyviai rinkti informaciją apie kibernetinėje erdvėje įvykdytą nusikaltimą, sumažinant tam reikalingą laiką ir rankinį darbą.

Siūloma ontologija leidžia skaitmeninės erdvės specialistui arba vartotojui suprasti pirminius teisinio skaitmeninių įrodymų nagrinėjimo srities terminus bei naujai aptinkamus terminus (terminijas), atitinkančius skirtingas operacines sistemas ir jų svarbą. Be to, sukurtoji ontologija yra techniškai patikrinta ir patvirtinta – ji yra nuosekli, išsami, glausta ir gali būti plėtojama. Naujų klasių, sąvokų ar jas atitinkančių apibrėžimų galima lengvai pridėti nekeičiant jau gerai sukurtoje ontologijoje suformuluotų apibrėžimų.

1.8. Rezultatų patvirtinimas

Disertacijos tema yra paskelbti šeši straipsniai ir viena santrauka. Du iš straipsnių atspausdinti moksliniuose žurnaluose, indeksuotuose „Web of Science“ duomenų bazėje. Keturi pranešimai ir viena santrauka paskelbti mokslinėse konferencijose Lietuvoje ir užsienyje.

1.9. Disertacijos struktūra

Disertaciją sudaro įvadas, trys pagrindiniai skyriai, išvados, literatūros sąrašas bei autoriaus publikacijų sąrašas. Bendra disertacijos apimtis – 122 puslapiai, įskaitant 41 paveikslėlį, 16 lentelių ir 139 literatūros šaltinius.

2. NAUJOVIŲ ANALIZĖ

Antrasis disertacijos skyrius „Naujovių analizė“ yra skirtas pagrindinių terminų apibrėžimui ir literatūros teisinio skaitmeninių įrodymų nagrinėjimo srityje naudojamų ontologijų ir vizualizavimo tema tyrimui. Literatūros tyrimas rodo, kad teisinio skaitmeninių įrodymų nagrinėjimo srityje yra naudojami įvairūs metodai, skirti laiko analizei, pvz., *CyberForensic TimeLab* [98], *Encase* [46], *Zeitline* [31], *Forensic Toolkit* [51], ECF [35] ir daugelis kitų. Dauguma jau sukurtų metodų sprendžia tik dvi teisinio skaitmeninių įrodymų nagrinėjimo srities problemas: automatinį įvykių atkūrimą ir didžiulio kiekio bei skirtingų duomenų šaltinių apdorojimą. Kai kurie metodai gali padėti spręsti kitus du svarbius klausimus, susijusius su duomenų vientisumu ir aiškiu naudojamo tyrimo modelio aprašymu. Vis dėlto šiais metodais nėra sprendžiama laiko analizės problema, ir jie nepadeda kibernetinės erdvės specialistams veiksmingai suprasti laiko juostos tyrimo procese [83, 39 ir 121]. Siekiant išspręsti šią problemą, disertacijoje siūlomas naujas metodas, paremtas abstrakcijos metodu. Šis metodas yra aptartas 3 skyriuje.

Įvairūs autoriai ar tyrėjai yra sukūrę virtinę ontologijų, skirtų teisinio skaitmeninių įrodymų nagrinėjimo sričiai, skirtų padėti skaitmeninės praktikos specialistams suprasti naujus terminus ir aktualijas, su kuriomis susiduriama atkuriant laiko juostas ir vykdant laiko analizę. Tarp pavyzdžių yra teisinio skaitmeninių įrodymų nagrinėjimo ontologija, skaitmeninių tyrimų ontologija (DIALOG), skaitmeninių įrodymų semantinė ontologija (DESO), teisinio skaitmeninių įrodymų nagrinėjimu pagrįsta ontologija (F-DOS) ir t. t. Visos šios ontologijos yra orientuotos į tikslą ir sukurtos konkretiems tikslams ir uždaviniams, iškylantiems konkrečiais atvejais ar konkrečiuose scenarijuose. Vis dėlto šios ontologijos negali būti taikomos įvairiausiems atvejams ar scenarijams. Esamos teisinio skaitmeninių įrodymų nagrinėjimo srities ontologijos taip pat negali padėti skaitmeninių įrodymų tyrėjui interpretuoti naujų terminų (terminijų), su kuriais susiduriama atliekant laiko analizę. Be to, literatūros tyrimas taip pat rodo, kad esamos ontologijos nėra techniškai patikrintos ir patvirtintos, todėl rengiant disertaciją buvo sukurta nauja ontologija, paremta abstrakcijos metodu. Ši ontologija yra aptarta 3 skyriuje.

Teisinio skaitmeninių įrodymų nagrinėjimo procesas gali būti palengvintas integruojant informacijos vizualizavimo metodus į esamą teisinio skaitmeninių įrodymų nagrinėjimo darbo eigos metodą, žinomą kaip EPIC procesas (angl. *Explore, Investigate and Correlate*). Teisinio skaitmeninių įrodymų nagrinėjimo srityje esama įvairių informacijos vizualizavimo metodų, tokių kaip geometrinė transformuotų vaizdų vizualizacija [34, 137 ir 131], piktogramų vaizdų vizualizacija [34, 17 ir 114], pikselinių vaizdų vizualizacija [116 ir 95], grafinė vizualizacija [48] ir t. t. Vizualizacijos technikos pasirinkimas labai priklauso nuo duomenų pobūdžio, nes jokia vizualizacijos technika netinka visų tipų duomenims. Atliktame tiriamajame darbe abstrakcija pagrįsto požiūrio, skirto laiko analizei, rezultatas yra pateikiamas tekstinių duomenų arba informacijos forma. Taigi, abstrakcija pagrįsto metodo rezultatams vizualizuoti naudojama grafika, pagrįsta vizualizacijos technika, aptarta 4 skyriuje.

3. TYRIMO PROJEKTAS IR METODAI

Trečiasis disertacijos skyrius „Tyrimo projektas ir metodai“ yra skirtas abstrakcija pagrįstos laiko analizės metodikos aprašymui ir siūlomai naujai ontologijai, paremtai abstrakcijos metodu, siekiant apibrėžti skirtingas terminijas, su kuriomis susiduriama analizuojant prietaisų su skirtingomis operacinėmis sistemomis laiko juostas.

3.1. Siūloma laiko analizės metodika

Tyrinėjimo procesas priklauso nuo informacijos, pateikiamos komandomis pagrįstų teisinio skaitmeninių įrodymų nagrinėjimo įrankių, t. y. *Log2timeline* ir *Psort, plaso* failo, pavidalu. Be to, *Psort* leidžia konvertuoti *plaso* failą į įprastus failų formatus, tokius kaip *L2TCSV*, t. y. *Log2timeline* kableliais atskirtas reikšmes. L2TCSV laiko juostoje yra 17 fiksuotų laukų, kaip nurodyta 1 lentelėje, taigi duomenis iš L2TCSV failo galima importuoti į *Excel* lapą.

Teisinio skaitmeninių įrodymų nagrinėjimo įrankiai *Log2timeline* ir *Psort* generuoja gerai struktūrizuotą laiko juostą, tačiau didžiulis duomenų kiekis ir tai, kad yra pateikiami visi įvykiai (net ir nereikšmingiausi), nustelbia svarbius lemiamus įvykius, turinčius didelę įtaką. Struktūrizuota laiko juosta turi būti transformuota, kad būtų galima surinkti skaitmeninius įrodymus atliekant laiko analizę. Sugeneruota struktūrizuota laiko juosta susideda iš daugelio elementų: duomenų, pateiktų iš skirtingų išteklių, kartojimo, duomenų kiekio, duomenų nevienalytiškumo ir t. t. Taigi visos šios problemos sudaro sudėtingą laiko juostą ir neleidžia skaitmeninės srities tyrėjams suprasti laiko juostos ir identifikuoti skirtingus vartotojo atliekamos veiklos tipus konkrečiame skaitmeniniame įrenginyje. Tyrėjai pasimeta daugybės įvykių sraute. Siekiant išspręsti šias problemas ir transformuoti struktūrizuotą laiko juostą, analizuojant struktūruotą laiko juostą sukurtas naujas požiūris, pagrįstas abstrakcijos koncepcija. Konkrečios kompiuterinės sistemos įvykių ir artefaktų laiko juostos analizė padeda sumažinti laiko juostos sudėtingumą, padalijant ją į skirtingus ir atitinkamus įvykių ir artefaktų laiko juostos lygius. Struktūrinė laiko juosta suskirstyta į keturis įvykių ir artefaktų laiko juostos lygius, t. y.: *Įvykiai: aukštas lygis* (nauji įrašai ir naršymas žiniatinklyje), *Įvykiai: žemas lygis* (naršymas žiniatinklyje, modifikavimo veiksmai), *Artefaktų lokacija: aukštas lygis* (įskaitant visas programas), ir *Artefaktų lokacija: žemas lygis*.

Pirmojo lygio laiko juosta, t. y. *Įvykiai: aukštas lygis* (nauji įrašai ir naršymas žiniatinklyje), vaizduoja informaciją, susijusią su daugybe veiklų, atliekamų žiniatinklyje ir sistemoje lokaliai, su dideliu abstrakcijos lygiu. Tai apima naršymą žiniatinklio puslapiuose, informacijos atsisiuntimą, laiškų kūrimą, vartotojo sukurtų failų įrašus ir pačią sistemą. Antrojo lygio laiko juosta, t. y. *Įvykiai: žemas lygis* (naršymas žiniatinklyje, modifikavimo veiksmai), iliustruoja išsamią informaciją, susijusią su žiniatinklio veikla, ir failų sąrašą, sukurtą vartotojo ir sistemos. Tai apima interneto puslapių, kuriuos pasiekia vartotojas, URL adresus, vartotojo pašto adresus, naudojamus rašant ir gaunant laiškus, failo dydį, tipą ir daug daugiau žemesnių abstrakcijos lygio elementų.

1 lentelė. L2TCSV failo laukai – įrankis *Psort* [16]

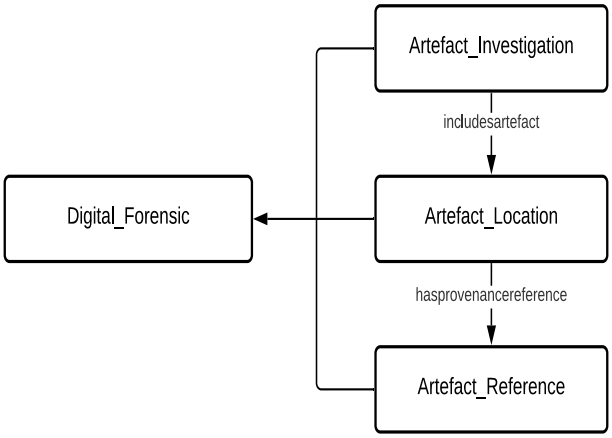
Laukas	Aprašymas
Date	Data, kada įvyko įvykis
Time	Laikas, kada įvykis įvyko
Timezone	Laiko juosta, kuri buvo naudojama įrankiui iškviesti
MACB	Modifikavimas, prieiga, kūrimas ir atsiradimas
Source	Šaltinio trumpasis pavadinimas, pvz., REG – registro įrašai
source type	Šaltinio aprašymas
Type	Laiko žymos tipas, pvz., paskutinis prisijungimas arba paskutinis įrašas
User	Koks vartotojo vardas yra susietas su įvykiu, jei toks yra
Host	Koks pagrindinio sistemos mazgo pavadinimas yra susietas su įrašu, jei toks yra

Short	Apima trumpą aprašymo lauką, kuriame saugomas tekstas
Desc	Čia saugoma didžioji dalis analizuojamos informacijos
version	Nurodo laiko žymos versijos numerį
Inode	Suteikia analizuojamo failo *Inode* numerį
Notes	Papildoma saugojimo vieta informacijai apie kai kuriuos įvesties modulius
format	Įvesties modulis, kuris buvo naudojamas analizei
Extra	Išanalizuota informacija, kuri sujungiama ir saugoma čia. Visa ši informacija sudaro laiko superjuostą, kurią sukuria *Log2Timeline*

Be to, trečiojo metodikos lygio laiko juosta, t. y. *Artefakto lokacija: aukštas lygis* (įskaitant visus programų failus), iliustruoja informaciją, kuri pateikiama pirmuose dviejuose metodikos lygiuose kartu su informacija, susijusia su įvairiomis vartotojo ir sistemos naudojamomis programomis atlikti įvairių tipų operacijas ar veiklą, t. y. lokaliai arba internetu. Galų gale, ketvirtojo metodikos lygio laiko juosta, t. y. *Artefakto lokacija: žemas lygis*, iliustruoja visą informaciją, kuri pateikiama pirmuosiuose trijuose metodikos lygiuose su žemiausiu abstrakcijos lygiu.

3.2. Naujoji ontologija, pagrįsta pasiūlyta metodika

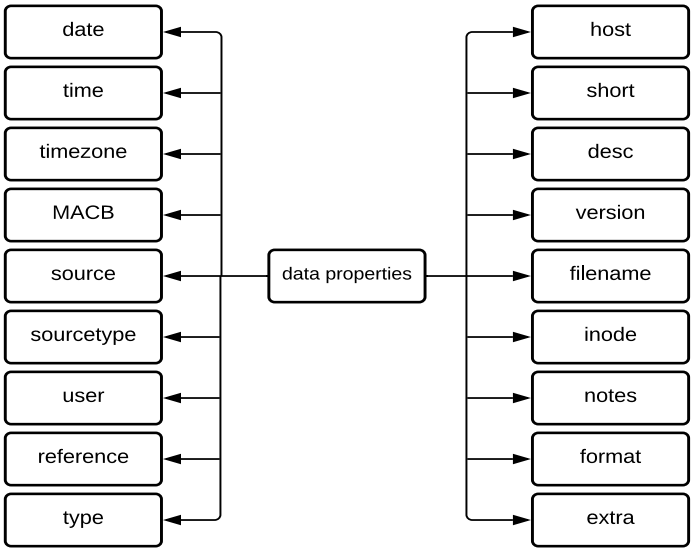
Skaitmeniniai teisinio skaitmeninių įrodymų nagrinėjimo įrankiai generuoja nestruktūrizuotas laiko juostas iš įvairių duomenų šaltinių. Nestruktūrizuotas laiko juostas sunku interpretuoti dėl kognityvinės perkrovos ir semantikos įvairovės. Taigi skaitmeninės srities specialistai negali suprasti naujai aptinkamų terminų ir jų surinkti. Norint išspręsti šias problemas, reikalingas perspektyvus metodas, kuriame būtų pateikiama teisinga ir patikima informacija, leidžianti vartotojui struktūrizuoti duomenis ir standartizuoti jų pateikimą. Šiuo tikslu buvo sukurtas teisinio



1 pav. Klasės ir objektų savybės [17]

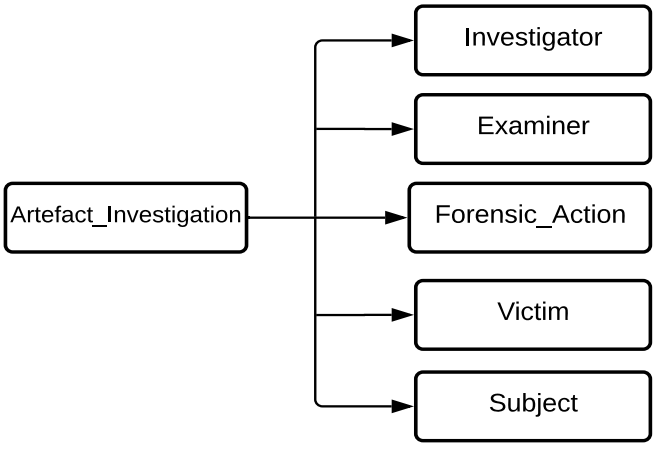
skaitmeninių įrodymų nagrinėjimo metodas, paremtas žinių modeliu – vadinamasis ontologinis metodas. Pagrindinės idėjos, kuriomis siekiama sukurti naują teisinio skaitmeninių įrodymų nagrinėjimo srities ontologiją, pirmiausia yra padėti praktikams suprasti naujas terminijas, su kuriomis susiduriama tyrimo metu, ir ryšius tarp jų. Antroji

ontologijos sukūrimo idėja yra dalijimasis srities (skaitmeninės teismo ekspertizės) žiniomis tarp mokslininkų, skaitmeninių praktikų ir vartotojų.



2 pav. Duomenų savybės [17].

Digital_Forensic yra pagrindinė klasė, atspindinti bendrą kriminalistikos srities koncepciją, kuriai yra kuriama ontologija, siekiant apibrėžti bendrą žodyną, dalytis informacija ir pakartotinai naudoti bei analizuoti srities žinias. Pagrindinėje klasėje yra apibrėžti trys poklasiai,



3 pav. *Artefact_Investigation* [17]

atspindintys konkretesnes skaitmeninės kriminalistikos srities sąvokas: *Artefact_Investigation* (Artefaktų tyrimas), *Artefact_Location* (Artefaktų lokacija) ir *Artefact_Reference* (Artefaktų nuoroda). Siekiant parodyti tarp šių klasių egzistuojantį ryšį, apibrėžiamos dvi objekto ypatybės, pvz., *includesartefact* ir *hasprovenancereference*, kaip parodyta 1 paveikslėlyje, o, siekiant parodyti ryšį tarp egzemplioriaus arba individo ir duomenų reikšmės, apibrėžta 18 duomenų savybių, kaip parodyta 2 paveikslėlyje [17].

Pirmasis poklasis *Artefact_Investigation* apibrėžia pagrindinius terminus, susijusius su teisinio skaitmeninių įrodymų nagrinėjimo procesu, o bylos tyrimo metu skirtingi specialistai įvairiais etapais turi atlikti skirtingus veiksmus. Ontologijoje kiekviena sąvoka apibūdinama apibrėžiant klasę. Taigi, norint apibrėžti poklasius, susijusius su poklasiu *Artefact_Investigation*, apibrėžiami nauji poklasiai, kaip parodyta 3 paveikslėlyje. Antrasis poklasis *Artefact_Location* reprezentuoja naująja abstrakcija pagrįstą laiko juostos analizės metodą ir apima keturis poklasius, atitinkančius keturis abstrakcijos metodo lygius, o būtent *Įvykiai: aukštas lygis, Įvykiai: žemas lygis, Artefaktų lokacija: aukštas lygis* ir *Artefaktų lokacija: žemas lygis*.

Trečiasis poklasis *Artefact_Reference* reprezentuoja skirtingų tipų duomenų šaltinius, su kuriais susiduriama atliekant laiko analizę. Šaltinių skaičius skiriasi priklausomai nuo operacinės sistemos, jos versijos bei duomenų kiekio. Vis dėlto šaltiniai padeda tyrėjui efektyviai atpažinti laiko juostoje esančią informaciją bei įvairius veiksmus, kuriuos vartotojas atlieka konkrečiame įrenginyje.

4. EKSPERIMENTINIAI TYRIMAI

Ketvirtasis skyrius „Eksperimentiniai tyrimai“ yra skirtas iliustruoti naudojant abstrakcija pagrįsto laiko analizės metodą padarytas išvadas, naująją ontologiją ir vizualizavimo techniką.

4.1. Išvados, padarytos taikant abstrakcija pagrįsto laiko analizės metodą

Siekiant iliustruoti naujojo abstrakcija pagrįsto metodo teikiamas laiko analizės galimybes, buvo atlikti keli eksperimentai, įvykdyti pasitelkiant objektinio programavimo kalbą, t. y. *Java*. Šiuo tikslu abstrakcija pagrįstas metodas buvo naudojamas įrenginiuose su įvairiomis operacinėmis sistemomis – *Windows, Android* ir *iOS*. Disertacijoje pateikiami dviejų kiekviena iš minėtų operacinių sistemų – *Windows, Android* ir *iOS* – pagrįstų laiko analizės atvejų tyrimai, siekiant parodyti, kad siūlomas metodas gali išanalizuoti laiko juostą, sugeneruotą teisinio skaitmeninių įrodymų nagrinėjimo įrankių *Log2timleline* ir *Psort*. Kadangi siūlomas abstrakcija pagrįstas laiko analizės metodas padalija sugeneruotą laiko juostą į keturis įvykių ir artefaktų lygius, kad būtų sudaryta atitinkama ir atpažįstama laiko juosta, todėl galutinis siūlomo metodo rezultatas taip pat yra keturi įvykių ir artefaktų laiko juostos lygiai. Kiekviename lygyje atsižvelgiama į skirtingą įvykių ir laukų skaičių, o, atliekant laiko analizę, įgyvendinama daugybė mechanizmų, siekiant sudaryti vartotojui ar skaitmeniniam specialistui suprantamą laiko juostą ir lengvai surinkti skaitmeninius įrodymus iš laiko juostos.

4.1.1. Operacinė sistema *Windows*

Disertacijoje nagrinėjami du su operacine sistema *Windows* susiję atvejai, t. y. atvejai, susiję su šaltiniais WEBHIST ir LNK. Pirmoji atvejo analizė, būtent šaltinio WEBHIST analizė,

atlikta siekiant pateikti aktualią ir organizuotą informaciją, susijusią su vartotojo veikla žiniatinklyje konkrečiame skaitmeniniame įrenginyje. 4 ir 5 paveikslėliuose pavaizduoti siūlomo metodo rezultatai keturių laiko juostos įvykių ir artefaktų, atitinkančių šaltinį WEBHIST, lygių forma. Kiekviename metodo lygmenyje atsižvelgiama į skirtingą laukų skaičių ir numatyti skirtingi mechanizmai lengvai suprantamai laiko juostai sudaryti. Skirtingi 6, 7, 10 ir 18 laukai nagrinėjami analizuojant *Įvykiai: aukštas lygis, Įvykiai: žemas lygis, Artefaktų lokacija: aukštas lygis* ir *Artefaktų lokacija: žemas lygis*, taip pat atskirų mechanizmų įgyvendinimas kiekviename lygyje, siekiant išspręsti kitas laiko juostos problemas. Pirmieji du metodo lygiai pateikia konkrečios vartotojo atliekamos veiklos

Events: High Level (Level 1)
Date: 11/14/2017
Time: 11:46:15
Source: WEBHIST
Short: https://mail.google.com Visit: mail
Reference: 257192

Events: Low Level (Level 2)
Date: 11/14/2017
Time: 11:46:15
Source: WEBHIST
Short: https://mail.google.com Visit: mail
Extra: https://mail.google.com/mail
Reference: 257192

Artefact Location: High Level (Level 3)
Date: 11/14/2017
Time: 11:46:15
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Short: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 201..
Visit: mail
Extra: https://mail.google.com/mail
Desc: sanxxxxxx525@gmail.com
Reference: 257192

Artefact Location: Low Level (Level 4)
Date: 11/14/2017
Time: 11:46:15
Timezone: UTC
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Type: Last Visited Time
User & Host: -
Short: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 201...
Desc: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 2017 For 380 Vacancies Apply Now - sanxxxxxx525@gmail.com - Gmail) [count: 0] Host: mail.google.com Type: [LINK - User clicked a link] (URL not typed directly - no typed count)
Version: 2
Filename: OS:C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\History
Inode & Notes: -
Format: sqlite/chrome_history
Extra: schema_match: False; sha256_hash: 582bcc588c7bc39ce0d789951fde1bd8296982a04d8a26eb09a582a901302ae3
Reference: 257192

4 pav. Operacinės sistemos Windows šaltinio WEBHSIT analizės rezultatas [17]

Keys:
URL: https://mail.google.com/mail/u/0/#inbox/15fba5850d593d64 (FCI Recruitment 2017 For 380 Vacancies Apply Now - sanxxxxxx525@gmail.com - Gmail)
Search Term: mail
Browser: Google Chrome
Description: LINK - User clicked a link

5 pav. Operacinės sistemos *Windows* šaltinio WEBHSIT analizės rezultatas [17]

apžvalgą; ji apima ir vartotojo naršomo tinklalapio adresą bei vartotojo lankomo tinklalapio tipą. Paskutiniai du metodo lygiai suteikia išsamesnę ir labai svarbią informaciją, kad būtų galima tiksliau suprasti konkrečią veiklą. Jie apima visą vartotojo naršomo tinklalapio adresą žiniatinklyje, žiniatinklio naršyklės, naudotos tinklalapiui pasiekti, pavadinimą ir prieigos prie konkretaus tinklalapio būdą. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą, susijusią su vartotojo vykdoma veikla žiniatinklyje, ir analizuoti informaciją, surinktus skaitmeninius įrodymus, kaip parodyta 4 paveikslėlyje.

Events: High Level (Level 1)
Date: 11/07/2017
Time: 20:11:35
Source: LNK
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Reference: 223117

Events: Low Level (Level 2)
Date: 11/07/2017
Time: 20:11:35
Source: LNK
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Extra: -
Reference: 223117

Artefact Location: High Level (Level 3)
Date: 11/07/2017
Time: 20:11:35
MACB: ...B
Source: LNK
Source type: Windows Shortcut
Short: D:\Doctorate Studies\chrome history window 7.txt
Visit: -
Extra: -
Desc: Empty description] File size: 57 File attribute flags: 0x00000020 Drive type: 3 Drive serial number: 0xc04f69f2 Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt
Reference: 223117

Artefact Location: Low Level (Level 4)
Date: 11/07/2017
Time: 20:11:35
Timezone: UTC
MACB: ...B
Source: LNK
Source type: Windows Shortcut
Type: Creation Time
User & Host: -
Short: [Empty description] D:\Doctorate Studies\chrome history window 7.txt
Desc: [Empty description] File size: 57 File attribute flags: 0x00000020 Drive type: 3 Drive serial number: 0xc04f69f2 Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt
Version: 2
Filename: OS:C:\Users\User\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\5f7b5f1e01b83767.automaticDestinations-ms
Inode & Notes: -
Format: olecf/olecf_automatic_destinations/lnk
Extra: birth_droid_file_identifier: 6df44ae9-c4d4-11e7-8ac6-a0afbdac1ec0; birth_droid_volume_identifier: a6ab9a4e-a31c-4e48-9416-b0cb2766758a; droid_file_identifier: 6df44ae9-c4d4-11e7-8ac6-a0afbdac1ec0; droid_volume_identifier: a6ab9a4e-a31c-4e48-9416-b0cb2766758a; sha256_hash: a39a0b9e3a0344d2feddf8168148344eb466887e864e8fcc0a276c559b3d11a7
Reference: 223117

6 pav. Operacinės sistemos *Windows* šaltinio LNK analizės rezultatas [17]

Keys:
Address and Name:
D:\Doctorate Studies\chrome history window 7.txt
Source type: Windows Shortcut
Description: Volume label: New Volume Local path: D:\Doctorate Studies\chrome history window 7.txt Link target: <My Computer> D:\Doctorate Studies\chrome history window 7.txt

7 pav. Operacinės sistemos *Windows* šaltinio LNK analizės rezultatas [17]

Antrasis atvejo tyrimas, o būtent operacinės sistemos *Windows* šaltinio LNK tyrimas, yra analizuojamas siekiant pateikti svarbios informacijos, susijusios su failais, kuriuos dažnai naudoja vartotojas. 6 ir 7 paveikslėliuose pavaizduoti siūlomo metodo rezultatai įvykių ir artefaktų laiko juostos keturių lygių forma, atitinkantys šaltinį LNK. Įvykiuose nagrinėjami skirtingi 6, 7, 10 ir 18 *Įvykiai*: *aukštas lygis, Įvykiai: žemas lygis, Artefaktų lokacija: aukštas lygis* ir *Artefaktų lokacija: žemas lygis*, kartu su skirtingų mechanizmų įgyvendinimu. Pirmieji du lygiai, t. y. *Įvykiai: aukštas lygis* ir *Įvykiai: žemas siūlomo metodo lygis* suteikia informacijos, susijusios su vartotojo daug kartų naudojamu failu, apžvalgą, pvz., pavadinimą (naršymo naudojant naršyklę *Chrome* istoriją, *Windows 7*), tipą (.txt) ir konkretaus failo vietą (D:\Doctorate Studies\). Be to, paskutiniai du laiko juostos lygiai, t. y. *Artefaktų lokacija: aukštas lygis* ir *Artefaktų lokacija: žemas lygis*, suteikia papildomos išsamios ir svarbios informacijos, susijusios su konkrečiu failu, kuris yra dažnai naudojamas vartotojo. Tai apima vartotojo su tuo failu atliekamų operacijų tipą, pvz., modifikavimą, atidarymą, pakeitimą arba sukūrimą. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą, susijusią su konkrečiu dažnai atidaromu failu, ir išanalizuoti informaciją, siekdamas surinkti skaitmeninius įrodymus, kaip parodyta 6 paveikslėlyje.

4.1.2. Operacinė sistema *Android*

Du operacine sistema *Android* pagrįstų laiko juostų atvejų tyrimai, susiję su šaltiniais META ir FILE, parodo naujo metodo galimybes. Panašiai kaip operacinė sistema *Windows*, operacinės sistemos *Android* atveju kiekviename metodo lygyje atsižvelgiama į tą patį laukų skaičių, o atitinkamai laiko juostai sudaryti numatyti tie patys mechanizmai. Įvykiuose nagrinėjami skirtingi laukai 6, 7, 10 ir 18: aukštas lygis, *Įvykiai: žemas lygis*, *Artefaktų lokacija: aukštas lygis* ir *Artefaktų lokacija: žemas lygis*, taip pat atskirų mechanizmų įgyvendinimas kiekviename lygyje, siekiant išspręsti kitas laiko juostos problemas. Pirmasis tyrimo atvejis, t. y. šaltinis META, analizuojamas siekiant pateikti informaciją, susijusią su skirtingais vartotojo atliktais veiksmais, susijusiais su failais ir aplankais, tokiais kaip Modifikavimas, Prieiga, Keitimas ir Gimimas (Sukūrimas). 8 ir 9 paveikslėliuose pateikiami siūlomo metodo rezultatai keturių abstrakcijos metodo lygių, atitinkančių šaltinį META, forma. Pirmasis metodo lygis, t. y. *Įvykiai: aukštas lygis*, suteikia glaustos informacijos, susijusios su konkrečiu vartotojo atliekamu veiksmu, susijusiu su failu ir aplanku. Tai apima konkretaus failo pavadinimą, tipą ir saugojimo vietą. *Įvykiai: žemas lygis* ir *Artefaktų lokacija: aukštas lygis* suteikia papildomos informacijos. Tai apima informaciją, kokias operacijas atlieka vartotojas su konkrečiu failu ar aplanku, failo turinį, pvz., žodžių skaičių, simbolių skaičių ir t. t., taip pat informaciją, kuri pateikiama skiltyje *Įvykiai: aukštas lygis*. Paskutiniame elemente *Artefakto lokacija: žemas lygis* pateikiama išsamesnė ir naudingesnė informacija, pvz., autoriaus vardas ir programos, naudojamos konkrečiai operacijai su failu atlikti, pavadinimas. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą ir analizuoti svarbią informaciją iš laiko juostos, susijusią su skirtingais

veiksmais ar operacijomis, kurias vartotojas atlieka naudodamas tam tikrą failą, pasitelkdamas konkrečias programas ir t. t., kaip parodyta 8 paveikslėlyje.

Events: high Level (level 1)
Date: 11/10/2015
Time: 12:46:00
Source: META
Short:
OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Reference: 77

Events: low Level (level 2)
Date: 11/10/2015
Time: 12:46:00
Source: META
Short:
OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Extra:
number_of_paragraphs:25
total_time:0
Reference: 77

Artefact location: high Level (level 3)
Date: 11/10/2015
Time: 12:46:00
MACB: M..B
Source: META
Source type: Open XML Metadata
Short:
OS:D:\yu\Download\Using the marketing mix to drive change.docx
Visit: -
Extra:
number_of_paragraphs:25
total_time:0
Desc: Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89
Reference: 77

Artefact location: low Level (level 4)
Date: 11/10/2015
Time: 12:46:00
Timezone: UTC
MACB: M..B
Source: META
Source type: Open XML Metadata
Type: Content Modification Time; Creation Time
User & Host: -
Short: Author: User
Desc: Creating App: Microsoft Office Word App version: 14.0000 Last saved by: Vartotojas Author: User Revision number: 2 Template: Normal Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89 Hyperlinks changed: false Links up to date: false Scale crop: false
Version: 2
Filename:
OS:D:\yu\Download\Using the marketing mix to drive change.docx
Inode & Notes: -
Format: Openxml
Extra: doc_security: 0; i4: 1; number_of_paragraphs: 25; sha256_hash: cd2d4ad6058b86d15c6fffcdb08cdd94deacdba41d7dc0397553ae0649f6aa59; shared_doc: false; total_time: 0
Reference: 77

8 pav. Operacinės sistemos *Android* šaltinio META analizės rezultatas [17]

Keys:
Address and Name: OS:D:\yu\Download\Using the marketing mix to drive change.docx
Operation type: M..B
Application: Microsoft Office Word
Description: Creating App: Microsoft Office Word App version: 14.0000 Last saved by: Vartotojas Author: User Revision number: 2 Template: Normal Number of pages: 5 Number of words: 1877 Number of characters: 10703 Number of characters with spaces: 12555 Number of lines: 89

9 pav. Operacinės sistemos *Android* šaltinio META analizės rezultatas [17]

Antrasis atvejo tyrimas, o būtent operacinės sistemos *Android* šaltinis FILE, yra analizuojamas siekiant pateikti svarbią informaciją, susijusią su failais, kurie yra prieinami konkrečiame mobiliajame įrenginyje su operacine sistema *Android*. 10 ir 11 paveikslėliuose parodytas siūlomo metodo rezultatas keturių įvykių ir artefaktų laiko juostos lygių forma. Panašiai kiekviename lygyje atsižvelgiama į tą patį laukų skaičių kartu su atitinkamos informacijos sudarymo mechanizmais, kaip nurodyta šaltinyje META.

Events: high Level (level 1)
Date: 12/07/2015
Time: 15:26:13
Source: FILE
Short:
D:\yu\SHAREit\audios\Tum_
Ho_Mera_Pyar-
K_K(DesiTape.Com).mp3
Visit: -
Reference: 86

Events: low Level (level 2)
Date: 12/07/2015
Time: 15:26:13
Source: FILE
Short:
D:\yu\SHAREit\audios\Tum_
Ho_Mera_Pyar-
K_K(DesiTape.Com).mp3
Visit: -
Extra: file_size: 2167277;
file_system_type: OS;
is_allocated: True
Reference: 86

Artefact location: high level (level 3)
Date: 12/07/2015
Time: 15:26:13
MACB: M...
Source: FILE
Source type: OS Content Modification Time
Short:
D:\yu\SHAREit\audios\Tum_H
o_Mera_Pyar-
K_K(DesiTape.Com).mp3
Visit: -
Extra:
file_size: 2167277;
file_system_type: OS;
is_allocated: True
Desc:
OS:D:\yu\SHAREit\audios\Tu
m_Ho_Mera_Pyar-
K_K(DesiTape.Com).mp3
Type: file
Reference: 86

Artefact location: low level (level 4)
Date: 12/07/2015
Time: 15:26:13
Timezone: UTC
MACB: M...
Source: FILE
Source type: OS Content Modification Time
Type: Content Modification Time
User: -
Host: -
Short:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Desc: OS:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3 Type: file
Version: 2
Filename: OS:D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Inode: -
Notes: -
Format: filestat
Extra: file_size: 2167277; file_system_type: OS; is_allocated: True; sha256_hash:
ae44d66a82fffb5a828ef83432eb147f564050355df62fe3ec6ba9cfa6908862
Reference: 86

10 pav. Operacinės sistemos *Android* šaltinio FILE analizės rezultatas [17]

Keys:
Address and Name:
D:\yu\SHAREit\audios\Tum_Ho_Mera_Pyar-K_K(DesiTape.Com).mp3
Operation type: M...
Application: SHAREit

11 pav. Operacinės sistemos *Android* šaltinio FILE analizės rezultatas [17]

Šiuo tyrimo atveju informacija apie konkretų failą pateikiama skirtingu laiko juostos lygiu su skirtingu detalumo abstrakcijos lygiu. Pirmieji dviejų lygių metodai pateikia glaustą informaciją, susijusią su konkrečiu prieinamu failu, pvz., failo pavadinimą, tipą, vietą ir dydį. Šiuo atveju failo pavadinimas, tipas, vieta ir dydis yra atitinkamai Tum_Ho_Mera_Pyar-K_K(DesiTape.Com), mp3(audio), D:\yu\SHAREit\audios\ ir apie 2.16 MB. Paskutiniai du metodo lygiai pateikia išsamesnę informaciją kartu su informacija, pateikta pirmuose dviejuose metodo lygiuose. Ji apima vartotojo arba pačios sistemos faile atliekamos operacijos tipą, operacijai atlikti naudojamą programą ir t. t. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą ir išanalizuoti atitinkamą laiko juostos informaciją, susijusią su konkrečiu failu, pasiekiamu operacinės sistemos *Android* pagrindu veikiančiuose mobiliuosiuose įrenginiuose, pvz., failo pavadinimą, tipą, vietą, dydį, su failu susijusią atliktą operaciją ir t. t., kaip parodyta 10 paveikslėlyje.

4.1.3. Operacinė sistema *iOS*

Disertacijoje aptarti du tyrimo atvejai, susiję su operacinės sistemos *iOS* laiko juostų analize, t. y. šaltinių WEBHSIT ir *iMessage* analizė. Kaip ir operacinių sistemų *Windows* bei *Android* atveju, kiekviename metodo lygyje atsižvelgiama į tą patį laukų skaičių, o atitinkamos laiko juostos sudarymui numatyti tie patys mechanizmai. Pirmosios atvejo analizės metu šaltinis WEBHIST yra analizuojamas siekiant pateikti atitinkamą informaciją apie žiniatinklio veiklą, kurią vartotojas atlieka žiniatinklyje. Pirmieji du metodo lygiai suteikia aukštesnį informacijos, susijusios su žiniatinklio veikla, abstrakcijos lygį. Tai apima konkretaus tinklalapio, kurį pasiekia vartotojas, adresą, tai, kaip šis konkretus tinklalapis pasiekiamas – ar vartotojas įveda adresą adreso juostoje arba spusteli nuorodą, bei vartotojo atliekama turinio paieška. Paskutiniuose dviejuose siūlomo metodo laiko juostos lygiuose pateikiama išsami informacija kartu su informacija, pateikta pirmuosiuose dviejuose lygiuose. Ji apima žiniatinklio naršyklės, kurią vartotojas naudojo žiniatinklyje operacijai atlikti, pavadinimą, vartotojo atliekamos žiniatinklio operacijos tipą ir t. t. Remdamasis siūlomo metodo rezultatais, specialistas gali nesunkiai suprasti laiko juostą ir analizuoti atitinkamą informaciją iš laiko juostos, susijusią su konkrečia vartotojo žiniatinklyje atliekama veikla. Tai apima vartotojo pasiekiamo tinklalapio adresą, žiniatinklio naršyklės, kurią vartotojas naudoja tam tikram tinklalapiui naršyti, pavadinimą, turinio paiešką tinklalapyje ir t. t., kaip parodyta 12 ir 13 paveikslėliuose.

Antrasis atvejo tyrimas, o būtent mobiliojo įrenginio su operacine sistema *iOS* šaltinio *iMessage* analizė, yra analizuojamas siekiant pateikti svarbią informaciją, susijusią su veikla, kuri atliekama lokaliai nenaudojant interneto, kaip parodyta 14 ir 15 paveikslėliuose. Pirmieji du naujojo metodo lygiai rodo tikrą informaciją, kuria dalijasi du vartotojai. Be to, paskutiniai du metodo lygiai suteikia gyvybiškai svarbios ir išsamios informacijos, susijusios su šia veikla. Tai apima bendrinamą informaciją, bendrinamos informacijos gavėją ir siuntėją, kontaktinį numerį, naudojamą informacijai siųsti arba gauti, ir t. t. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą ir analizuoti atitinkamą laiko juostos informaciją, susijusią su konkrečia vietine veikla, kurią vartotojas atlieka įrenginyje su operacine sistema *iOS*. Tai apima faktinę komunikaciją, vykusią tarp dviejų vartotojų, komunikacijoje naudoto vartotojo kontaktinį numerį ir t. t., kaip parodyta 14 paveikslėlyje, iš laiko juostos, susijusią su konkrečia vartotojo žiniatinklyje atliekama veikla. Tai apima vartotojo pasiekiamo tinklalapio adresą, žiniatinklio naršyklės, kurią vartotojas naudoja tam tikram tinklalapiui naršyti, pavadinimą, turinio paiešką tinklalapyje ir t. t., kaip parodyta 12 ir 13 paveikslėliuose.

Events: high level (level 1)
Date: 10/31/2019
Time: 08:16:41
Source: WEBHIST
Short: https://www.kaunokolegija.lt
Visit: LINK 1
Reference: 2024

Events: low level (level 2)
Date: 10/31/2019
Time: 08:16:41
Source: WEBHIST
Short: https://www.kaunokolegija.lt
Visit: LINK 1
Extra: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Reference: 2024

Artefact location: high level (level 3)
Date: 10/31/2019
Time: 08:16:41
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Short: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Visit: LINK 1
Extra: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Desc: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies) [count: 1] Host: www.kaunokolegija.lt Visit Source: [SOURCE_SYNCED] Type: [LINK - User clicked a link] (type count 1 time)
Reference: 2024

Artefact location: low level (level 4)
Date: 10/31/2019
Time: 08:16:41
Timezone: UTC
MACB: .A..
Source: WEBHIST
Source type: Chrome History
Type: Last Visited Time
User & Host: -
Short: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies)
Desc: https://www.kaunokolegija.lt/ (Kaunas College - modern and practical studies) [count: 1] Host: www.kaunokolegija.lt Visit Source: [SOURCE_SYNCED] Type: [LINK - User clicked a link] (type count 1 time)
Version: 2
Filename: OS:D:\applebackup\acf4b9617ef493f11fa0dd4e11bce6cd6eb5b3f2\fa\faf971ce92c3ac508c018dce1bef2a8b8e9838f1
Inode & Notes: -
Format: sqlite/chrome_history
Extra: schema_match: False; sha256_hash: 1ec938e2eed7efe16719dde363cf12efd3fc7201e1c995f54ce8d18fb55c497b
Reference: 2024

12 pav. Operacinės sistemos *iOS* šaltinio WEBHIST analizės rezultatas [17]

Keys:
URL: https://www.kaunokolegija.lt
Search Term: Kaunas College - modern and practical studies
Browser: Google Chrome
Description: : LINK - User clicked a link

13 pav. Operacinės sistemos *iOS* šaltinio WEBHIST analizės rezultatas [17]

Events: high level (level 1)
Date: 18/01/2020
Time: 09:45:17
Source: iMessage
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Reference: 39925

Events: low level (level 2)
Date: 18/01/2020
Time: 09:45:17
Source: iMessage
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Extra: -
Reference: 39925

Artefact location: high level (level 3)
Date: 18/01/2020
Time: 09:45:17
MACB: ...B
Source: iMessage
Source type:
Apple iMessage Application
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Visit: -
Extra:-
Desc: iMessage ID: +370xxxxxxx4 Read Receipt: True Message Type: Received Service: SMS Message Content: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Reference: 39925

Artefact location: low level (level 4)
Date: 18/01/2020
Time: 09:45:17
Timezone: UTC
MACB: ...B
Source: iMessage
Source type: Apple iMessage Application
Type: Creation Time
User & Host: -
Short:
Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Desc: iMessage ID: +370xxxxxxx4 Read Receipt: True Message Type: Received Service: SMS Message Content: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Version: 2
Filename:
OS:D:\applebackup\acf4b9617ef493f11fa0dd4e11bce6cd6eb5b3f2\3d\3d0d7e5fb2ce288813306e4d4636395e047a3d28
Inode & Notes: -
Format: sqlite/imessage
Extra: schema_match: False; sha256_hash:
78570d1699f93d2ccd80cdf525568b6133aa8084e153dac1c774d34d8cf8fc1e
Reference: 39925

14 pav. Operacinės sistemos *iOS* šaltinio iMessage analizės rezultatas [17]

Keys:
iMessage: Good morning. I can't do 9am tomorrow I have interviews. What about Wednesday?
Application: Apple iMessage

15 pav. Operacinės sistemos *iOS* šaltinio *iMessage* analizės rezultatas [17]

Antrasis atvejo tyrimas, o būtent mobiliojo įrenginio su operacine sistema *iOS* šaltinio *iMessage* analizė, yra vertinamas siekiant pateikti svarbią informaciją, susijusią su veikla, kuri atliekama lokaliai nenaudojant interneto, kaip parodyta 14 ir 15 paveikslėliuose. Pirmieji du naujojo metodo lygiai rodo tikrą informaciją, kuria dalijasi du vartotojai. Be to, paskutiniai du metodo lygiai suteikia gyvybiškai svarbios ir išsamios informacijos, susijusios su šia veikla. Tai apima bendrinamą informaciją, bendrinamos informacijos gavėją ir

siuntėją, kontaktinį numerį, naudojamą informacijai siųsti arba gauti, ir t. t. Remdamasis siūlomo metodo rezultatais, specialistas gali lengvai suprasti laiko juostą ir analizuoti atitinkamą laiko juostos informaciją, susijusią su konkrečia vietine veikla, kurią vartotojas atlieka įrenginyje su operacine sistema *iOS*. Tai apima faktinę komunikaciją, vykusią tarp dviejų vartotojų, komunikacijoje naudoto vartotojo kontaktinį numerį ir t. t., kaip parodyta 14 paveikslėlyje.

4.1.4. Palyginimas

Palyginamasis tyrimas analizuojamas atsižvelgiant į 21 skirtingą veiksnį arba požymį, įvertinant siūlomą metodą bei penkis esamus metodus, t. y. metodus, kuriuos sukūrė Guðjónsson [59], Inglot ir Liu [70], Soltani, Seno ir Yazdi [124], Esposito ir Peterson [47] ir Hargreaves ir Patterson [62]. Šios savybės reiškia, kad tam tikru metodu rekonstruota laiko juosta susideda iš svarbios informacijos, susijusios su įvairiomis vartotojo operacijomis ar veiksmais, atliekamais konkrečiuose skaitmeniniuose įrenginiuose, informacijos vientisumu, iš informacijos, susijusios su prieinamomis programomis, failais ir sistemos failais, pašto adresų, vartotojo naudotų komunikacijai ar kitiems tikslams, identifikavimu, operacinės sistemos nevienalytiškumu ir t. t., kaip parodyta 10 lentelėje. Rezultatai rodo (žr. 2 lentelę), kad siūlomas abstrakcija pagrįstas metodas gali padėti efektyviau ir tinkamiau išanalizuoti laiko juostą nei esami kiti metodai, kadangi siūlomas metodas apima visas (21-ą) savybes. Kita vertus, daugelis šių savybių nebūdingos esamiems metodams. Be to, siūlomas metodas suteikia išsamios informacijos, susijusios su visais veiksmais, atliekamais prisijungus ir neprisijungus prie interneto, ar susijusios su turimais vartotojo ir sistemos failais, vartotojo naudojamomis programomis, el. pašto adresais ir kontaktiniais numeriais, naudotais komunikacijai ar kitiems tikslams, ir t. t., kartu nustatant unikalią struktūrą kiekviename lygyje, eliminuojant įvykių ir artefaktų pasikartojimą, pašalinant nereikšmingą informaciją, kad vartotojas arba skaitmeninės erdvės tyrėjai galėtų sudaryti atitinkamą ir atpažįstamą laiko juostą.

2 lentelė. Siūlomo abstrakcija pagrįsto ir esamų metodų palyginimas

Savybės	Mūsų metodas	Guðjónsson [59]	Inglot & Liu [70]	Soltani, Seno & Yazdi [124]	Esposito & Peterson [47]	Hargreaves & Patterson [62]
1. Nuorodinis numeris, naudojamas ryšiui su pirminiu šaltinio failu palaikyti.	+	-	-	-	-	-
2. Pasikartojimų ir nereikšmingų detalių pašalinimas.	+	-	-	-	-	-
3. Įvykių ir artefaktų laiko juostos padalijimas.	+	-	-	-	-	-
4. Unikalios struktūros identifikavimas kiekviename lygyje.	+	-	-	-	-	-
5. MAC adreso identifikavimas.	+	-	-	-	-	-

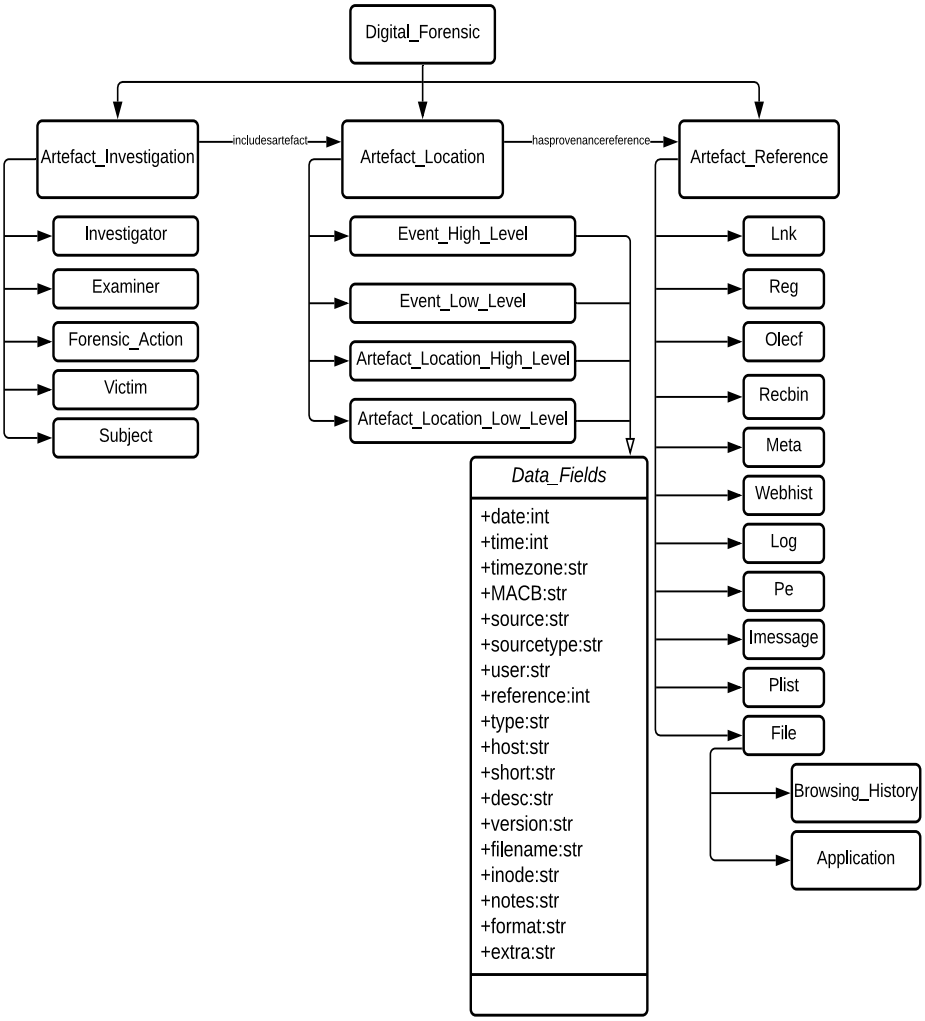
6. Įvairių vartotojų ir pačios sistemos naudojamų programų identifikavimas.	+	+	-	+	+	-
7. Komunikacijai naudojamų pašto adresų identifikavimas.	+	-	-	+	+	-
8. Informacijos, susijusios su nebepasiekiamais failais ir programomis, rinkimas.	+	+	-	-	-	-
9. Informacijos, susijusios su palaikomomis programomis ir sistemų failais, rinkimas .	+	+	-	+	-	-
10. Viso vartotojo pasiekiamų tinklalapių adresų identifikavimas.	+	+	+	+	+	+
11. Kaip (NUORODAS, TIPAS, FORMOS PATEIKIMAS, INTERNETO SERVERIS) pasiekiamas konkretus tinklalapis.	+	-	+	+	-	-
12. Faktinės vartotojo žiniatinklyje ieškotos informacijos rinkimas.	+	+	+	+	+	+
13. Įvairių interneto naršyklių, kurias vartotojas naudoja veiklai žiniatinklyje atlikti, sąrašas.	+	+	-	+	+	+
14. Informacijos, susijusios su failais arba programomis, kurios atsisiunčiamos iš interneto, rinkimas.	+	-	-	-	+	-
15. Informacijos, susijusios su failais ir programomis, kurias vartotojas pasiekia dažniausiai, rinkimas.	+	+	-	-	-	-
16. Informacija, susijusi su visų tipų failais ir	+	+	-	+	+	-

programomis.						
17. Informacija, susijusi su skirtingų tipų operacijomis, kurias vartotojas atlieka naudodamas konkretų failą, t. y. MACB.	+	-	-	+	-	-
18. Faktinės informacijos (*imessage*), kuria dalijasi vartotojai, rinkimas.	+	-	-	-	-	-
19. Komunikacijai ir kitiems tikslams naudojamų vartotojų kontaktinių numerių paieška.	+	-	-	-	-	-
20. Operacinės sistemos heterogeniškumas.	+	+	-	-	-	-
21. Duomenų vientisumas.	+	+	-	+	+	-

4.2. Naujosios ontologijos pranašumai

Naujai sukurta ontologija yra pagrįsta abstrakcija pagrįstu laiko juostos analizės metodu ir įgyvendinama naudojant ontologijos redaktoriaus *Protégé 5.5.0 Build* versiją *Beta-9* kartu su vizualizacijos papildiniais, o būtent *OWLViz, OntoGraf* ir VOWL, ir ontologijos vizualizacijos įrankį *OWLGred* [82] [17]. Pagrindinės teisinio skaitmeninių įrodymų nagrinėjimo srities sąvokos jau paaiškintos 3 skyriuje „Tyrimo projektas ir metodai“.

Šios pagrindinės sąvokos buvo apibrėžtos formuluojant ontologiją, kurioje sąvokos ir subsąvokos paaiškintos naudojant klases, poklasius ir savybes. Analizuojant įrenginių su skirtingomis operacinėmis sistemomis – *Windows, Android*, ir *iOS*, – laiko juostas, susiduriama su įvairiais naujais terminais. Taigi naujoji ontologija taip pat turėtų apimti informaciją apie tokias naujai aptinkamas terminijas. Šiam tikslui apibrėžiami nauji poklasiai, atitinkantys kiekvieną siūlomos naujosios ontologijos poklasyje *Artefact_Reference* aptinkamą terminologiją



16 pav. Siūloma ontologija

Rezultatai rodo, kad, analizuojant įrenginių su operacinėmis sistemomis *Windows, Android* ir *iOS* laiko juostas, susiduriama su vienuolika naujų terminų. Šios terminijos įtrauktos į siūlomą ontologiją apibrėžiant naują poklasį, atitinkantį kiekvieną naują terminą *Artefact_Reference* poklasyje, kaip parodyta 16 paveikslėlyje. Be to, 3 lentelėje pateikiama šių naujai pasitaikančių terminų aprašomoji informacija. Rezultatai rodo, kad operacinės sistemos *Windows* laiko juostoje vartojami ne daugiau nei devyni nauji terminai, o būtent „LNK“, „REG“, „OLECF“, „RECBIN“, „META“, „WEBHIST“, „LOG“, „PE“ ir „FILE“. Operacinės sistemos *Android* laiko juostoje vartojami keturi terminai, o būtent OLECF, „META“, „PE“ ir „FILE“. Operacinės sistemos *iOS* laiko juostoje vartojami šeši nauji terminai, o būtent „META“, „PE“, „WEBHIST“, „FILE“, „IMESSAGE“ ir „PLIST“.

3 lentelė. Naujos terminijos ir jų aprašymas [17]

Nr.	Terminijos (Subklasės)	Aprašymas
1.	WEBHIST	Informacija, susijusi su naršymo veikla, apimanti interneto puslapių, kuriuos atsidaro vartotojas, adresus, vartotojo pašto adresus, atsisiųstus failus ir taikomąją programą (žiniatinklio naršyklės pavadinimą), naudojamą prieigai prie internetinių paslaugų.
2.	RECBIN	Informacija apie failus, kuriuos vartotojas ištrynė iš sistemos ir šiukšlinės.
3.	PE	PE reiškia *Portable Executable*. PE formatuoti failai apima .exe, .apk, .ipa .dll ir .sys (tvarkyklės failus).
4.	FILE	Pateikiama su konkrečiu failu susijusi informacija, tokia kaip pavadinimas, tipas, vieta ir dydis.
5.	META	*Meta* dažnai apibūdinama kaip „duomenys apie duomenis“. Tai apima konkretaus dokumento ar failo modifikavimo, prieigos, keitimo ir atsiradimo (sukūrimo) informaciją.
6.	LNK	Nuorodiniai failai, susieti su programa arba failu, dažniausiai randami vartotojo a darbalaukyje arba visoje sistemoje ir baigiasi plėtiniu .LNK. Labai naudingi norint pasiekti failus, kurių sistemoje nebėra.
7.	REG	Informacija apie naudojamas programas ir .DAT failus, kurie yra skirti tik programoms palaikyti.
8.	OLECF	OLECF reiškia objektų susiejimo ir įterpimo sudėtinį failą. Jis apima .msp, .msi, .asd ir .automaticdestination-ms failus, kuriuose pateikiama informacija apie operacinės sistemos *Windows* ir kitų programų naujinimus.
9.	LOG	Išsaugo informaciją apie įvykius, vykstančius operacinėje sistemoje ar kitose programose.
10.	IMESSAGE	Teikia informaciją, susijusią su visais gautais ir siunčiamais pranešimais naudojant programą *Apple iMessage*.
11.	PLIST	Reiškia *Property List* ir yra programos duomenų saugojimo formatas.

4.2.1. Naujosios ontologijos įvertinimas

Naujoji ontologija, kuri remiasi abstrakcija pagrįstu laiko juostos analizės metodu, yra įvertinama naudojant du metodus: ontologijos taksonomijos įvertinimą ir ontologijos turinio įvertinimą, skirtą naujai ontologijai patikrinti ir patvirtinti. Patikrinimo procesui naudojamas ontologijos taksonomijos įvertinimo metodas, leidžiantis pateikti toliau nurodytus rezultatus (žr. 4 lentelę).

4 lentelė. Naujosios ontologijos įvertinimas

1. Nenuoseklumas	**I. Apskritiškumo klaida:** klaidų nėra, nes naujojoje ontologijoje jokia klasė nėra įvardijama kaip savo pačios konkretizavimas ar apibendrinimas.
	II. Skaidinių klaida: vertinant siūlomą naująją ontologiją, skaidinių klaidų nerasta. Sukurtoje ontologijoje nėra neteisingo atskiriamų klasių aprašymo, t. y. sukurtosios ontologijos klasė nėra išvestinė daugiau nei vienos klasės klasė.
	III. Semantinė klaida: sukurtoji ontologija yra be semantinių klaidų, nes visos ontologijos klasės ar sąvokos yra klasifikuojamos semantiškai teisingai.
2. Neišbaigtumas	**I. Neišsami sąvokų klasifikacija:** naujojoje ontologijoje yra vartojamos visos pagrindinės teisinio skaitmeninių įrodymų nagrinėjimo srities sąvokos. Ontologija taip pat apima abstrakcijos metodo terminus, vartojamus laiko juostos analizei, ir naujas terminijas, atitinkančias įrenginius su skirtingomis operacinėmis sistemomis (*Windows, Android* ir *iOS*).
	II. Skaidinių klaidos: sukurtojoje ontologijoje nėra skaidinių klaidų.
3. Pertekliškumas	**I. Gramatinis pertekliškumas:** sukurtojoje ontologijoje jokia klasė neturi daugiau nei vieną apibrėžimą ar aprašymą. Taigi naujoji ontologija yra be gramatinio pertekliškumo klaidų.
	II. Identiškas formalus kai kurių klasių ir egzempliorių apibrėžimas: sukurtoji ontologija neapima identiškų formalių kai kurių klasių ir egzempliorių apibrėžimų, nes joje nėra apibrėžimo, atitinkančio daugiau nei vieną klasę ir egzempliorius.

Patvirtinimui yra naudojamas ontologijos turinio įvertinimo metodas, kurio rezultatai yra pateikti toliau (žr. 5 lentelę).

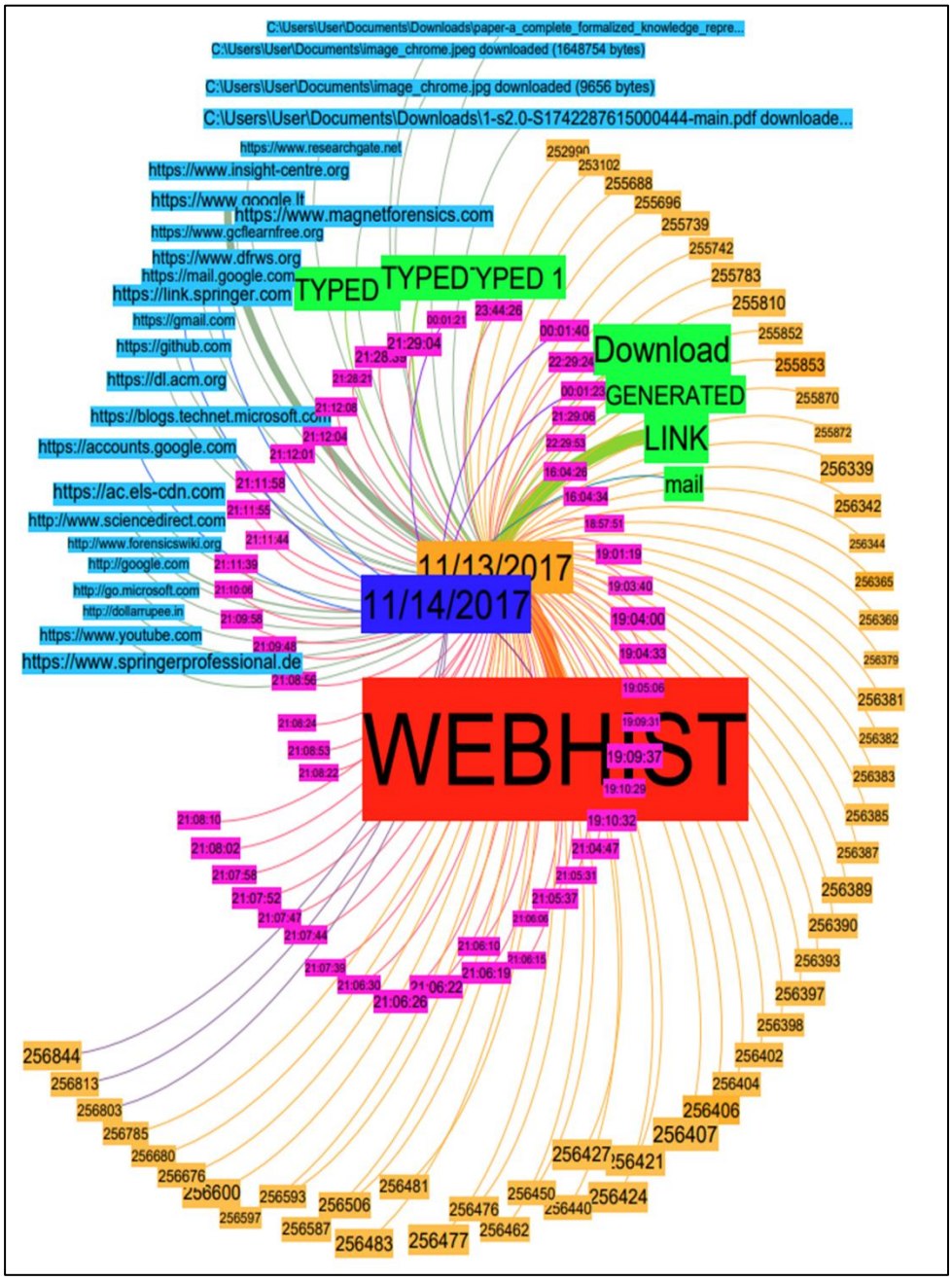
5 lentelė. Naujosios ontologijos įvertinimas

1. Nuoseklumas	Naujoji ontologija yra nuosekli, nes kiekvienas klasės apibrėžimas yra nuoseklus, atspindintis realų pasaulį ir apibrėžimas negeneruoja prieštaringos informacijos.
2. Išbaigtumas	Sukurtoji ontologija apima visas pagrindines teisinio skaitmeninių įrodymų nagrinėjimo srities terminijas bei naujus terminus, atitinkančius įrenginius su skirtingomis operacinėmis sistemomis, ir terminus, atitinkančius laiko juostos analizės abstrakcijos metodą. Be to, ontologijoje taip pat yra pateikti kiekvienos klasės sąvokos apibrėžimai. Taigi siūloma ontologija yra visiškai išbaigta.
3. Glaustumas	Siūloma ontologija yra glausta, nes joje nėra nereikšmingų klasių ar sąvokų apibrėžimų ar apibrėžimų kartojimosi.

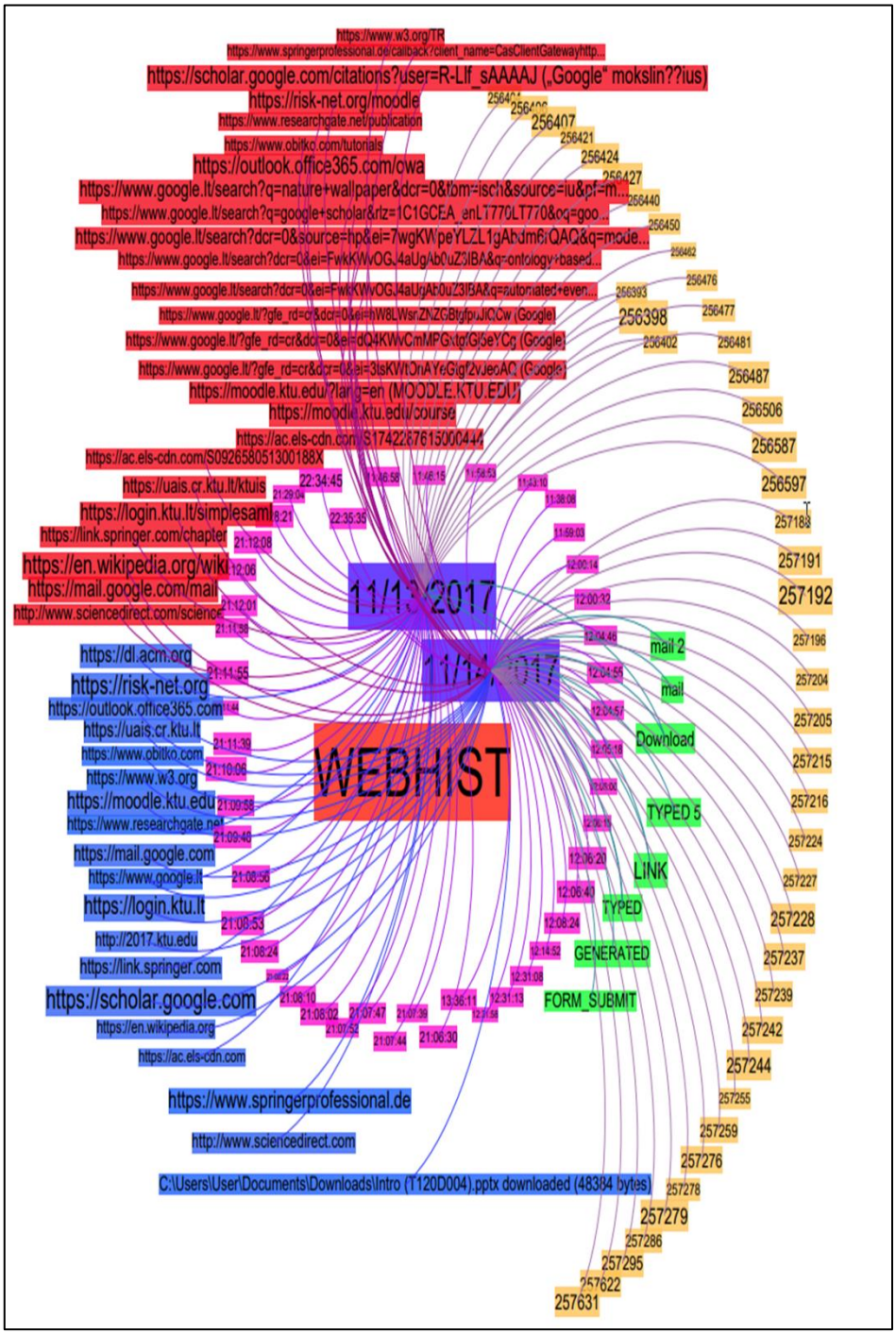
4. Išplečiamumas	Naujas klases ar sąvokas ir jas atitinkančius apibrėžimus galima lengvai pridėti nekeičiant jau gerai apibrėžtų apibrėžimų sukurtojoje ontologijoje. Taigi tai reiškia, kad galima lengvai plėsti odontologiją. Pavyzdžiui, naują klasę ir ją atitinkantį apibrėžimą galima lengvai įtraukti į naujosios ontologijos poklasį *Artefact_Reference*, nekeičiant kitų sąvokų.
5. Jautrumas	Naujoji ontologija yra lengvai plečiama. Tai reiškia, kad nedideli kai kurių apibrėžimų pakeitimai nepakeis kruopščiai suformuluotų klasių ar koncepcijos. Taigi sukurtoji ontologija nėra jautri pakeitimams.

4.3. Vizualizavimo rezultatai

Šiame skyriuje paaiškinta laiko juostos vizualizacija, parengta taikant naująjį abstrakcija pagrįstą metodą. Vizualizacija įgyvendinama vaizduojant naujojo abstrakcija pagrįsto metodo rezultatus, naudojant programinės įrangos *Gephi* versiją 0.9.2 201709241107. Naujojo abstrakcija pagrįsto metodo, skirto laiko juostos analizei, rezultatas yra pateikiamas tekstinės informacijos forma, kaip parodyta 3 skyriuje. Taigi naujojo metodo rezultato, susijusio su daugybe skirtingų vartotojo konkrečiame skaitmeniniame įrenginyje atliekamų veiklų, neįmanoma tinkamai pavaizduoti vienu atitinkamu paveikslėliu tuo pat metu. Šiuo tikslu vizualizacija įgyvendinama pavaizduojant rezultatus, pasiektus naudojant naująjį metodą, grafiškai iliustruojant daugybę veiklų atitinkančią laiko juostą. Laiko juostos atvejo analizė, išanalizuota taikant abstrakcija pagrįstą metodą, t. y. operacinės sistemos *Windows* šaltinį WEBHIST, išnagrinėta naudojant grafinės vizualizacijos techniką. Pagrindinis vizualizacijos technikos naudojimo siekis – parodyti vizualizacijos naudą, leidžiančią lengviau atpažinti ir vizualiai pateikti informaciją, susijusią su daugeliu vartotojo atliekamų veiklų, kurią sunku pasiekti teksto formato duomenų ar informacijos forma. Be to, žmogaus smegenys gali lengviau ir greičiau interpretuoti informaciją, gautą per vaizdines užuominas, palyginti su rašytine kalba. 17 paveikslėlyje parodyta įvykių vizualizacija: aukšto lygio abstrakcija pagrįstas laiko juostos analizės metodas, atitinkantis šaltinį WEBHIST. Grafinės vizualizacijos rezultatai leidžia labai lengvai ir akimirksniu peržiūrėti ir interpretuoti didelį kiekį informacijos – gerokai didesnį duomenų kiekį nei gaunamą iš tekstinės informacijos. Skirtingi daugiakampio formos mazgų ir kraštų dydžiai ir spalvos bei dvigubo apskritimo išdėstymo funkcijos naudojami labai vizualizuotai ir suprantamai pateikiant informaciją. Šiuo tyrimo atveju lygį *Įvykiai: aukštas lygis* atitinkanti vizualizacija labai aiškiai parodo skirtingus vartotojo atliekamus veiksmus internete ar žiniatinklyje kartu su informacija, susijusia su šiais veiksmais. Vartotojas gali greitai peržiūrėti ir interpretuoti informaciją, pvz., vartotojo



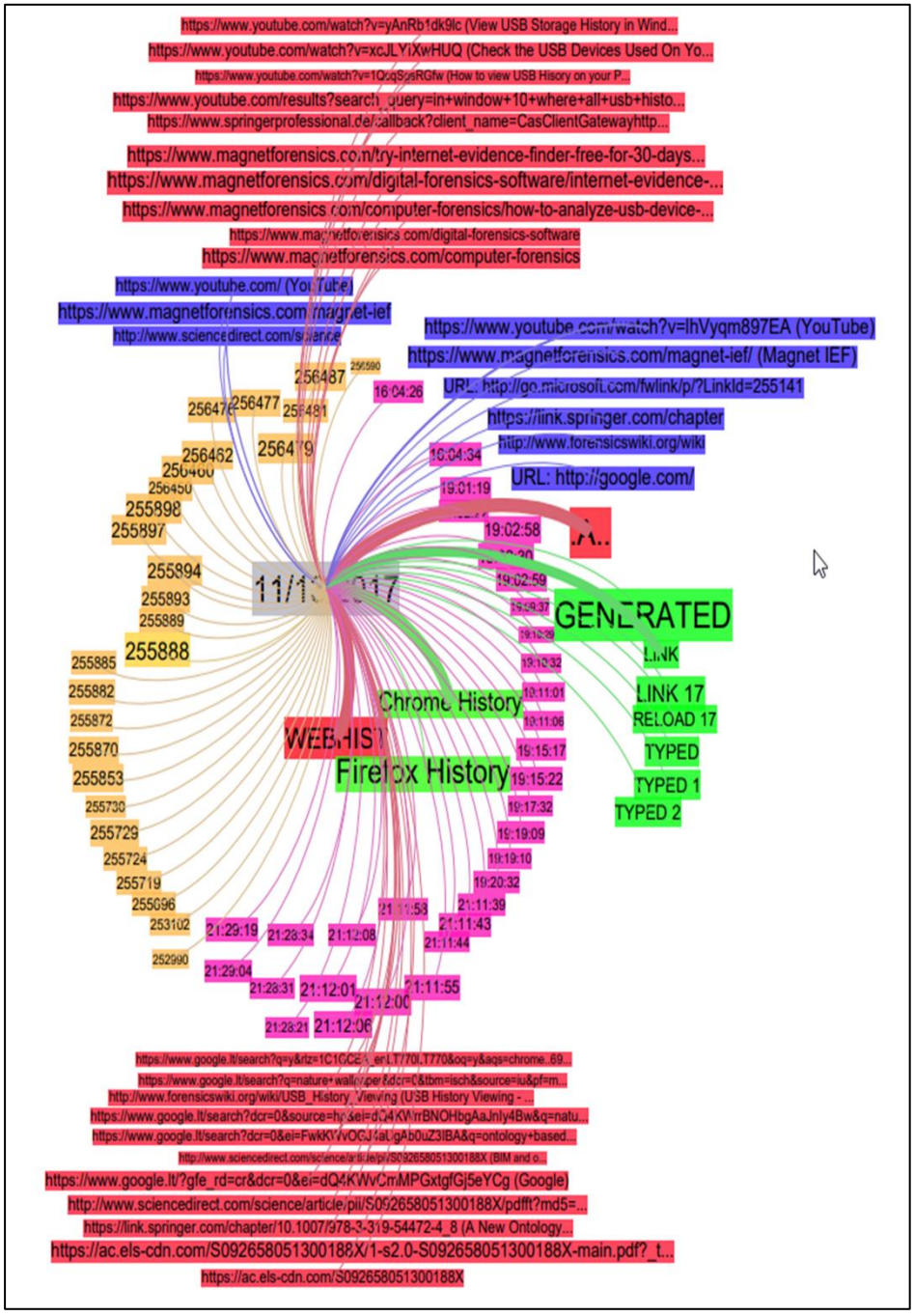
17 pav. *Įvykiai: aukštas lygis* vizualizacijos



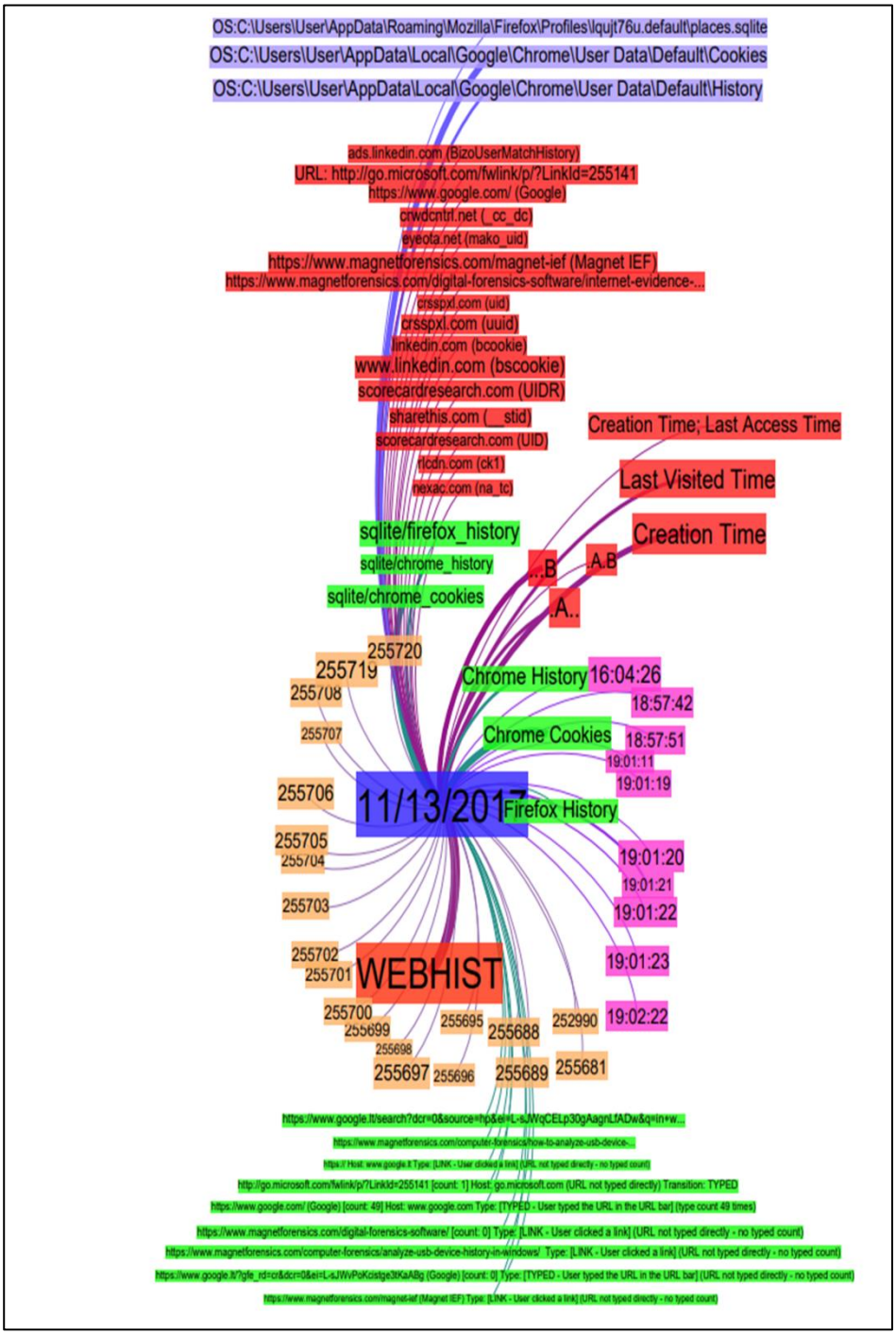
18 pav. *Įvykiai: žemas lygis* vizualizacijos

pasiekiamų tinklalapių pavadinimus, iš interneto atsisiųstos informacijos saugojimo vietas, tai, kokias operacijas vartotojas atlieka žiniatinklyje, veiksmų datą ir laiką ir daugelį kitų dalykų, kuriems nustatyti prireiktų daugiau laiko, kai informacija pateikiama tekstinės informacijos forma, palyginti su grafine vizualizacija. Pavyzdžiui, atidarytų tinklalapių pavadinimuose yra http://go.microsoft.com, https://www.magnetforensics, http://www.forensicswiki.org, http://www.sciencedirect.com, iš žiniatinklio atsisiųstos informacijos saugojimo vietos adresas apima C:\Users\User\Documents\image_chrome.jpg, atsisiųsta 9656 baitų, :\Users\User\Documents\image_chrome.jpeg atsisiųsta 1648754 baitų, ir kitos informacijos, aiškiai pastebimos ir pavaizduotos mėlynais daugiakampio formos mazgais. Panašiai 18 paveikslėlyje taip pat pateikta abstrakcija pagrįsto laiko juostos analizės metodo lygio *Įvykiai: žemas lygis*, atitinkančio WEBHIST, vizualizacija. Grafais pagrįstos vizualizacijos rezultatai pateikia labai aiškią ir suprantamą didesnio detalumo laipsnio informaciją. Joje pateikiami visi vartotojo pasiekiamų tinklalapių adresai, kas suteikia daugiau galimybių suprasti tikslią informacijos paiešką internete, pvz., https://www.researchgate.net/publication,https://www.obitko.com/tutorials, https://moodle.ktu.edu/?lang=en(MOODLE.KTU.EDU, https://www.google.lt/search?dcr=0&ei=FwkKWvOGJ4aUgAb0uZ3IBA&q=ontology+based ir t. t., kaip parodyta raudonos spalvos daugiakampio formos mazgais. Abstrakcija pagrįsto laiko juostos analizės metodo, atitinkančio šaltinį WEBHIST, lygio *Artefakto lokacija: aukštas lygis* grafais pagrįsta vizualizacija yra parodyta 19 paveikslėlyje. Šio lygio vizualizacija suteikia išsamesnės ir prasmingesnės informacijos, susijusios su įvairiais vartotojo žiniatinklyje atliktais veiksmais, kartu su informacija, pateikta grafais pagrįstoje ankstesnių dviejų abstrakcinio metodo lygių vizualizacijoje, t. y. lygių *Įvykiai: žemas lygis* ir *Įvykiai: aukštas lygis*. Pavyzdžiui, žiniatinklio naršyklės, kurią vartotojas naudoja atskiroms operacijoms žiniatinklyje atlikti, pavadinimas ir tai, kokios vartotojo atliekamos operacijos yra iliustruotos MACB, t. y. analizėje „modifikavimas, pasiekiamumas, keičiamumas ir sukūrimas (parengimas)“. Šiuo tyrimo atveju vartotojas įvairiai veiklai žiniatinklyje atlikti dažnai naudojo dvi skirtingas žiniatinklio naršykles, t. y. *Google Chrome* ir *Mozilla Firefox*. Šios žiniatinklio naršyklės ir operacijos (t. y. prieiga) yra aiškiai iliustruotos grafais pagrįstoje lygio *Artefakto lokalizacija: aukštas lygis* vizualizacijoje (žr. 19 pav.), atitinkamai žalios spalvos daugiakampio formos mazgais ir raudonos spalvos daugiakampio formos mazgais.

20 paveikslėlyje parodyta grafais pagrįstos lygio *Artefakto lokalizacija: žemas lygis* vizualizacijos rezultatas: žemo lygio abstrakcija pagrįstas laiko juostos analizės metodas, atitinkantis šaltinį WEBHIST. Grafais pagrįsta šio lygio vizualizacija suteikia išsamesnės ir aiškesnės informacijos nei ankstesni trys abstrakcija pagrįsto metodo lygiai. Pateikiami vartotojo pasiekiamų tinklalapių adresai ir išsami informacija apie tai, kaip vartotojas pasiekia šiuos tinklalapius. Pavyzdžiui, https://www.google.com/ (Google) [suskaičiuota: 49] serveris: www.google.com ĮVESTAS ADRESAS: [ĮVESTA – vartotojas įvedė URL adreso eilutėje] (simbolių, paspaustų įvedant adresą, skaičius – 49), https://www.magnetforensics.com/magnet-ief (Magnet IEF) ADRESO ĮVEDIMAS: [NUORODA – vartotojas paspaudė nuorodą] (URL neįvestas tiesiogiai – paspausti simboliai nesuskaičiuoti), https://www.magnetforensics.com/computer-forensics/analyze-usb-device-history-in-windows/ adreso įvedimas: [NUORODA – vartotojas paspaudė nuorodą] (URL neįvestas tiesiogiai – paspausti simboliai nesuskaičiuoti), https:// Host: www.google.lt Type:



19 pav. *Artefaktų lokacija: aukštas lygis* vizualizacijos



20 pav. *Artefaktų lokacija: žemas lygis* vizualizacijos

[NUORODA – vartotojas paspaudė nuorodą] (URL neįvestas tiesiogiai – paspausti simboliai nesuskaičiuoti) ir t. t. Ši informacija yra aiškiai matoma ir interpretuojama bei rodoma žaliais daugiakampio formos mazgais. Be to, šiame lygmenyje taip pat yra prieinama daugiau informacijos, susijusios su MACB operacijomis, pvz., failo sukūrimo laikas, apsilankymo laikas ir prieigos laikas (informacija aiškiai parodyta raudonos spalvos daugiakampio formos mazgais). Be to, šiame lygyje taip pat pateikiama informacija, susijusi su duomenų baze, kurią naršyklės naudoja vidiniams duomenims, pvz., naršyklės istorijai, saugoti.

Be to, 6 lentelė taip pat įtraukta siekiant iliustruoti išsamią abstrakcija pagrįsto metodo informacijos vizualizavimo įvairiais lygiais apžvalgą, naudojant grafais pagrįstą vizualizacijos techniką, atitinkančią šaltinį WEBHIST. Tai padeda vartotojui pasirinkti konkrečią vizualizaciją iš galimų vizualizacijų, atitinkančių keturis abstrakcija pagrįsto metodo laiko juostos analizės lygius, atsižvelgiant į vartotojo poreikius vizualizuoti informaciją ir tiksliau suprasti laiko juostą.

6 lentelė. Grafais pagrįsta vizualizacija

Grafais pagrįstos vizualizacijos technika				
Vizualizuota informacija	**Įvykiai: aukštas lygis**	**Įvykiai: žemas lygis**	**Artefaktų lokacija: aukštas lygis**	**Artefaktų lokacija: žemas lygis**
Įvykio ar veiksmo pavadinimas (WEBHIST)	✓	✓	✓	✓
Data ir laikas	✓	✓	✓	✓
Tinklalapio adresas (abstraktus)	✓	✓	✓	✓
Nuorodinis numeris	✓	✓	✓	✓
Veiksmų tipas (atsisiuntimas, el. paštas, NUORODA)	✓	✓	✓	✓
Saugojimo vieta	✓	✓	✓	✓
Tinklalapio adresas (išsamus)	×	✓	✓	✓
Žiniatinklio naršyklės informacija	×	×	✓	✓
Operacijos informacija (MACB)	×	×	✓	✓
Papildoma informacija, susijusi su atliktais veiksmais (NUORODA, ĮVESTA)	×	×	×	✓
Papildoma informacija, susijusi su naudojamomis interneto naršyklėmis	×	×	×	✓

5. IŠVADOS

1. Mokslinių publikacijų analizė parodė, kad esami laiko juostos rekonstravimo metodai nesudaro galimybės efektyviai išspręsti klausimų, susijusių su svarbiais skaitmeninio tyrimo proceso veiksniais, tokiais kaip įvykių identifikavimas, nevienalytiškumas ir didžiulis duomenų kiekis, tiksliai apibrėžtas tyrimo modelis, analizės galimybės, informacijos vientisumas ir lankstumas. Literatūros tyrimai taip pat parodė, kad prieinamos teisinio skaitmeninių įrodymų nagrinėjimo srities ontologijos neapima pagrindinių ir naujų terminų, atitinkančių įrenginius su įvairiomis operacinėmis sistemomis. Šios ontologijos yra sukurtos siekiant konkrečių tikslų ir uždavinių. Be to, šios ontologijos taip pat nėra techniškai patikrintos ir patvirtintos.
2. Atsižvelgiant į literatūros tyrimo rezultatus, sukurtas naujas abstrakcija pagrįstas laiko juostos analizės metodas. Šis metodas leidžia analizuoti laiko juostas, laiko juostos sudėtingumą sumažinant per laiko juostos padalijimo į keturis atitinkamus įvykių ir artefaktų lygius procesą.
3. Naujoji ontologija sukurta atsižvelgiant į esamų teisinio skaitmeninių įrodymų nagrinėjimo srities ontologijų literatūros tyrimo rezultatus. Sukurtoji ontologija padeda skaitmeninės erdvės tyrėjams suprasti pagrindinių teisinio skaitmeninių įrodymų nagrinėjimo srities terminų ir naujų terminų, atitinkančių įrenginius su įvairiomis operacinėmis sistemomis, reikšmes vienu metu. Be to, sukurtoji ontologija yra techniškai įvertinta.
4. Suformuluotasis abstrakcija pagrįstas metodas yra sukurtas vartojant objektinio programavimo kalbą, t. y. *Java*, ir gali būti naudojamas įrenginiuose su trimis skirtingomis operacinėmis sistemomis – *Windows, Android* ir *iOS* – eksperimentinių tyrimų metu. Eksperimentinių tyrimų išvados rodo, kad siūlomas metodas gali sumažinti teisinio skaitmeninių įrodymų nagrinėjimo srities įrankių generuotų laiko juostų sudėtingumą ir sprendžia pagrindines skaitmeninių tyrimų problemas. Be to, atliktas lyginamasis sukurtojo laiko juostos analizės metodo ir esamų laiko juostos rekonstrukcijos metodų tyrimas, pagrįstas 21 skirtingu veiksniu ar savybe. Lyginamojo tyrimo išvados rodo, kad, taikant naująjį metodą, galima efektyviau ir tinkamiau išanalizuoti laiko juostas nei naudojant esamus metodus.
5. Naujoji ontologija sukurta naudojant ontologijos redaktorių, t. y. *Protégé 5.5.0 Build* versiją *Beta 9*, kartu su įvairiais vizualizacijos papildiniais. Sukurtoji ontologija apima pagrindinę klasę „Teisinis skaitmeninių įrodymų nagrinėjimas“ ir tris poklasius, t. y. *Artefact_Investigation* (Artefaktų tyrimas), *Artefact_Location* (Artefaktų lokacija) ir *Artefact_Reference* (Artefaktų nuoroda). „*Artefact_Investigation*“ iliustruoja pagrindinius teisinio skaitmeninių įrodymų nagrinėjimo srities terminus, *Artefact_Location* poklasis parodo naują abstrakcija pagrįstą laiko juostos analizės metodą, o *Artefact_Reference* pateikia naujus terminus, atitinkančius įrenginius su skirtingomis operacinėmis sistemomis, su kuriais susiduriama atliekant laiko juostos analizę. Ontologija taip pat apima dvi objekto savybės ir 18 duomenų savybių. Laiko juostų analizės rezultatai rodo, kad, analizuojant įrenginių su operacinėmis sistemomis *Windows, Android* ir *iOS* laiko juostas, susiduriama su vienuolika naujų terminų. Be to, sukurtoji ontologija taip pat patikrinta ir patvirtinta atitinkamai naudojant ontologijos taksonomijos vertinimo metodą ir ontologijos turinio vertinimo metodą.
6. Siūlomo laiko juostų analizės metodo rezultatas yra pateikiamas tekstinės informacijos forma. Taigi rezultatų, susijusių su daugybe vartotojo atliekamų veiklų,

neįmanoma pavaizduoti vienoje figūroje ar paveikslėlyje. Šiuo tikslu vizualizavimo technika įgyvendinama remiantis siūlomo metodo rezultatais, siekiant grafiškai iliustruoti daugybę veiklų atitinkančią informaciją. Vizualizacijos rezultatas parodė, kad vartotojas vos iš vienos rezultato vizualizacijos gali gauti ir interpretuoti tikslią informaciją, susijusią su keliomis vartotojo vykdomomis veiklomis konkrečiame skaitmeniniame įrenginyje per tą patį laiko vienetą.

PUBLIKACIJŲ SĄRAŠAS

Straipsniai, paskelbti „Web of Science“ duomenų bazėje, turintys citavimo indeksą:

3. BHANDARI, S. & V. JUSAS. Abstrakcija pagrįstas teisinio skaitmeninių įrodymų nagrinėjimo srities laiko juostų rekonstravimo metodas. *Symmetry*, 2020, vol. 12, nr. 1.
4. BHANDARI, S. & V. JUSAS. Ontologija, pagrįsta *Log2timeline* ir *Psort* laiko juosta, parengta naudojant abstrakcijos metodą teisinio skaitmeninių įrodymų nagrinėjimo srityje. *Symmetry*, 2020, vol. 12, nr. 4.

Straipsniai, paskelbti „Web of Science“ duomenų bazėje, neturintys citavimo indekso:

6. BHANDARI, S. & V. JUSAS. Fazėmis pagrįstas teisinio skaitmeninių įrodymų nagrinėjimo srities laiko juostų rekonstravimo metodas. *2020 International conference on innovations in intelligent systems and applications (INISTA)*, Novi Sad, Serbia, 2020, pp. 1-6.
7. BHANDARI, S. & A. KULIKAJEVAS. Ontologija pagrįstas vaizdų atpažinimas: apžvalga. *IVUS 2018: proceedings of the international conference on information technologies*, Kaunas, Lithuania, 2018, vol. 2145, pp. 13-18.
8. BHANDARI, S. & V. JUSAS. Auditas: teisinis skaitmeninių įrodymų nagrinėjimas. *Proceedings of IASTEM international conference*, Krakow, Poland, 2019, pp. 7-11.
9. BHANDARI, S. & V. JUSAS. Teisinio skaitmeninių įrodymų nagrinėjimo srities laiko juostų analizės tobulinimas. *Proceedings of IASTEM international conference*, Krakow, Poland, 2019, pp. 1-6.
10. BHANDARI, S. & V. JUSAS. Duomenų tyrimas ir klasifikavimas teisinio skaitmeninių įrodymų nagrinėjimo srityje. *10th international workshop on data analysis methods for software systems*, Druskininkai, Lithuania, 2018, pp. 11.

APDOVANOJIMAI:

Dalyvavo ir apdovanotas KTU konkurse „Aktyviausi 2020 m. doktorantai“.

INFORMACIJA APIE DISERTACIJOS AUTORIŲ

Išsilavinimas:

2010–2013 m. I. K. Gudžralo Pandžabo technikos universiteto informacinių technologijų bakalauras.
2013–2016 m. I. K. Gudžralo Pandžabo technikos universiteto kompiuterių mokslo ir inžinerijos technologijų magistras.
2017–2022 m. Kauno technologijos universiteto informatikos inžinerijos doktorantūros studijos.

UDK 004:343.985(043.3)
SL 344. 2022-*-*, * leidyb. apsk. l. Tiražas 14 egz. Užsakymas * .
Išleido Kauno technologijos universitetas, K. Donelaičio g. 73, 44249 Kaunas
Spausdino leidyklos „Technologija“ spaustuvė, Studentų g. 54, 51424, Kauna