Towards Designing Shared Digital Forensics Instructional Materials

Weifeng Xu
School of Criminal Justice
College of Public Affairs
University of Baltimore
Maryland, USA
wxu@ubalt.edu

Lin Deng
Department of Computer and Information Sciences
Towson University
Maryland, USA
ldeng@towson.edu

Dianxiang Xu
Department of Computer Science and Electrical Engineering
University of Missouri at Kansas City
Missouri, USA
dxu@umkc.edu

***Abstract*—This paper presents a systematic approach to designing a series of digital forensics instructional materials to address the severe shortage of active learning materials in the digital forensics community. The materials include real-world scenario-based case studies, a set of hands-on problem-driven labs for each case study, and an integrated forensic investigation environment. In this paper, we first clarify some fundamental concepts related to digital forensics, such as digital forensic artifacts, artifact generators, and evidence. We then re-categorize knowledge units of digital forensics based on the artifact generators for measuring the coverage of learning outcomes and topics. Finally, we utilize a real-world cybercrime scenario to demonstrate how knowledge units, digital forensics topics, concepts, artifacts, and investigation tools can be infused into each lab through active learning. The repository of the instructional materials is publicly available on GitHub. It has gained nearly 600 stars and 22k views within several months.**

***Index Terms*—digital forensics, instructional materials, real-world case studies, artifacts, artifact generators**

I. Introduction

With the exponential increase of cybercrimes in recent years, the need for digital forensics expertise is growing quickly [1]. Many local, state, and federal law enforcement agencies (e.g., the FBI) and business entities rely on digital forensics professionals to identify malicious activities, reconstruct crime scene, and catch criminals. The Bureau of Labor Statistics reported that the demand for digital forensics-related jobs is expected to grow by 28 percent from 2016 to 2026. At this extraordinarily fast rate of growth, more than 28,000 jobs are expected to be added during that period [2]. The main task of digital forensics professionals is the recovery and investigation of digital evidence found in various digital devices. Qualified professionals need to have in-depth knowledge and solid experience of digital forensic evidence identification, acquisition, and examination, as well as presenting and explaining digital forensic evidence in courts.

Despite the urgent need, there are major barriers to cultivating digital forensics professionals to comprehend the core knowledge of digital forensics and to practice cyber investigation techniques and skills. These barriers include the lack of experienced educators and the severe shortage of publicly available hands-on digital forensics instructional materials, which posts significant challenges to digital forensics education community.

The paper proposes a systematic approach to design a series of digital forensics instructional materials for faculty, students, and the digital forensics community. These materials include eight complex real-world case studies, an integrated investigating environment, and a series of hands-on labs for each case study. In the approach, we first clarify some fundamental concepts related to digital forensics, such as digital forensic artifacts, artifact generators, and evidence. We then re-categorize knowledge units of digital forensics based on the artifact generators for measuring the coverage of learning outcomes and topics. Finally, we have utilized one real-world cybercrime scenario to demonstrate the approach of infusing knowledge units, digital forensics topics, concepts, artifacts, and investigation tools into labs through active learning.

The contributions of the paper include:

- Provided a fundamental understanding of how digital evidence is generated and what to investigate.
- Developed scenario-based and problem-driven hands-on labs with visualized solution guidelines for self-paced and enhanced active learning experience.
- Created an integrated Linux-based investigating environment in which students can practice labs and learn core concepts of digital forensics and cyber investigations.
- Published instructional materials on GitHub[1], so that everyone in the digital forensics community can share, use, and contribute.

The rest of the paper is outlined as follows: Sections II defines fundamental terms related to digital forensics, including artifacts, artifact generators, and evidence, and Section III describes generator-based Knowledge Units. Section IV discusses the design principles of the instructional materials. Section V shows the implemented instructional materials and feedback from digital forensics communities. Finally, Sections VI and VII summarize the related work and conclude this paper.

II. Digital Forensic Artifacts, Artifact Generators, and Evidence

The goal of the cybercrime investigation is to prove whether a suspect is a responsible person for an underlying crime by

[1]https://github.com/frankwxu/digital-forensics-lab

978-1-6654-8810-5/22/$31.00 ©2022 IEEE
DOI 10.1109/COMPSAC54236.2022.00025

reconstructing the crime scene. Figure 1 shows an evidence-driven approach to reconstructing a crime scene. It shows the traceability among a set of fundamental concepts used in cyber investigations, including suspects, activities, devices, artifacts, evidence, and investigators.

Since digital forensics practitioners and researchers do not often use standard terminology in their work [3], to facilitate our discussion, we provide formal definitions for the aforementioned concepts.

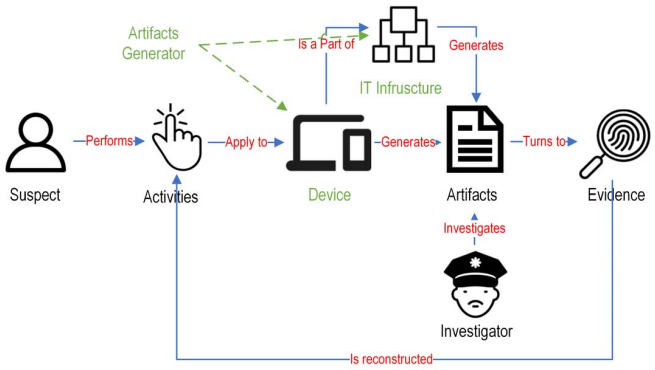


Fig. 1. An evidence-driven approach to reconstruct a crime scene.

Definition 1: Digital Forensic Artifacts: A digital forensic artifact, or simply called an artifact shown in Figure 1, is any type of item produced by a digital device, stored in an electronic form, and used for forensic investigations.

A digital forensic artifact is a by-product of a suspect's activities when using digital devices. Common digital artifacts include Word documents, pictures, applications, network traffic, and logs. Identifying digital forensic artifacts relevant to a crime is the first and fundamental step of digital forensics investigations.

Definition 2: Digital Forensic Artifact Generator (DFAG): A digital forensic artifact generator, or simply called an artifact generator, as shown in Figure 1, is a hardware or software that generates a digital forensic artifact. For example, when a file (i.e., a digital forensic artifact), is modified by a user, a corresponding Operating System (OS), i.e., a DFAG, will generate the last-modified timestamp and record it in the file metadata.

Technically, digital forensic artifacts are generated from (1) Information technology (IT) infrastructure that is required to operate and manage enterprise IT environments including hardware, operating systems, and networking. (2) Applications that are required to support business or personal objectives. We have categorized DFAGs based on the required IT infrastructure and installed applications. The types of DFAGs are listed as follows:

- Computer. It refers to the computer clients in a traditional client-server infrastructure architecture.
- Server. It refers to the servers in the traditional client-server architecture. It includes web servers, file servers, cloud storage, etc.
- Mobile and IoT devices. As the name indicates, it includes mobile and IoT devices, such as Android and iOS equipped smart devices, Alexa Echo, drones, etc.
- Networking device. It refers to network hardware, such as routers and switches, that enables network operations, management, and communication among systems.
- Operating system (OS). OS is responsible for managing system resources and hardware. It generates system-level artifacts.
- Application. It refers to the software applications installed in digital devices.

Definition 3: Digital Forensic Evidence: A digital forensic evidence, or simply called evidence shown in Figure 1, is a type of digital forensic artifact that is presentable for the purpose of proving a crime. Digital forensic evidence must be admissible [4] at court. Admissible evidence must satisfy some essential criteria, such as evidence relevance and authenticity. These criteria require evidence to have properties that are associated with the fundamental questions related to cyber investigations, such as who the suspect is, when an incident happened, what and how it happened. To answer these questions, the properties of evidence must include the artifacts from which evidence was extracted, evidence timestamps, devices and activities that generate the evidence, the tools that are used to obtain the evidence, and the investigator who analyzes evidence.

III. DFAG-based Knowledge Units

A knowledge unit (KU) is a thematic collection that includes several related educational topics [5]. The concept of KU is widely adopted by the Center of Academic Excellence (CAE), sponsored by the National Security Agency (NSA). Educators use KUs to outline major topics and learning outcomes in their instructional materials, aiming to optimize the clarity, straightforwardness, and conformity. This section introduces DFAG-based KUs and compare them with CAE-Cyber Defense (CAE-CD) KUs.

A. Types of DFAG-based Knowledge Units

KUs that correspond to the DFAG are called as DFAG-based KUs. While a DFAG generates artifacts because of the usages of devices, a DFAG-based KU defines outcomes and learning topics directly associated with the artifacts to reconstruct the usages of devices. For example, Computer DFAG generates artifacts related to computers. Computer Forensics, the corresponding KU, is to provide students with the ability to apply forensic techniques to investigate and analyze artifacts generated by the computer DFAG. In other words, DFAG defines what artifacts will be generated and KU defines what artifacts need to be covered in the instructional materials to perform investigation tasks. We define two types of DFAG-based KUs:

- Device-specific KUs. They are derived from the hardware of DFAGs. Device-specific KUs include Computer Forensics, Server Forensics, Mobile and IoT forensics, and Network Device forensics.

- Domain-specific KUs. A device in an IT infrastructure cannot work properly without storage media, memory, operating system, application software, and network data. Domain-specific KUs refine device-specific KUs to knowledge domains that crosscut all devices. Domain-specific KUs include Media Forensics, Memory Forensics, Operating System Forensics, Software Forensics, and Network Forensics.

B. The coverage comparison of DFAG-based KUs and CAE-CD KUs

CAE-CD, a de facto guideline for Academic Excellence in Cybersecurity, provides a list of KUs for Digital Forensics. CAE-CD implicitly defines two device-specific KUs. i.e., Host Forensics and Device Forensics, and two domain-specific KUs, i.e., Media Forensics and Network Forensics. Each KU is specified using the following pattern:

The intent of the [KU name] Knowledge Unit is to provide students with the ability to apply forensic techniques to investigate and analyze [the DFAG of the KU].

The content of [] varies based on specific KU and its corresponding DFAG. For example, the description of Host Forensics is defined as "the intent of the [Host Forensics] Knowledge Unit is to provide students with the ability to apply forensics techniques to investigate and analyze [a host in a network]". Table I shows the coverage comparison between DFAG-based KUs and KUs defined by CAE-CD. Specifically, we make the following adjustments:

- Split Host Forensics defined in CAE-CD into Computer Forensics and Server Forensics. Although computers and servers share many commonalities, they have many distinct characters.
- Change Device Forensics to Mobile and IoT Forensics. We specify devices like Mobile and IoT devices because of the popularity of Mobile and IoT devices.
- Add Memory Forensics. The analysis of volatile data in a computer's memory dump is critical to investigate and identify attacks or malicious behaviors that do not leave easily detectable tracks on hard drive data.
- Add Operating System Forensics. An operating system (OS) generates system-level artifacts. Both memory and operating system forensics focus on the investigation of live operating systems.
- Add Software Forensics: Analyzing software source code or compiled code is essential to comprehend applications' behaviors.

TABLE I
THE COVERAGE COMPARISON OF DFAG-BASED KUS AND CAE-CD KUS

KU Types	**DFAG-based KUs**		**CAE-CD KUs**	
	Name	***DFAG of KU***	***Name***	***DFAG of KU***
Device-specific	Computer Forensics	a computer	Host Forensics	a host in a network
	Server Forensics	a server in network		
	Mobile & IoT Forensics	mobile & IoT devices	Device Forensics	a device
	Network Device Forensics	a network device		
Domain-specific	Media Forensics	no changes	Media Forensics	a particular media
	Network Forensics	no changes	Network Forensics	network traffic
	Memory Forensics	a device memory		
	OS Forensics	an operating System		
	Software Forensics	software applications		

IV. ACTIVE LEARNING-ORIENTED DESIGN PRINCIPLES

Active learning creates excitement in the classroom [6]. To infuse active learning into the design of our materials, and to enhance students' learning experience, we define the following design principles for the active learning digital forensics instructional materials:

- **Real-world scenarios**: Labs are developed based on well-known real-world cybercrimes, so that students can have experience on what actually happened in real-world.
- **Comprehensive labs**: Cover as many digital forensics KUs and topics as possible.
- **Problem-driven labs**: Each lab consists of a list of step by step technical questions that guide students to the end.
- **Self-paced learning**: These instructional materials come with detailed introduction and are designed for digital forensics communities, including students and faculty with diverse backgrounds.
- **An integrated open-source lab environment**: The investigation environment is built based on Linux system and includes all the necessary digital forensics tools.

A. Real-world scenarios

The real-world case study is an effective way to define the objectives of digital investigations and boost students' interest. Table II shows nine real-world cybercrime cases used in the instructional materials. Each case study consists of a list of labs. Disks and memory involved in these cases are acquired and publicly available as well on GitHub. NIST Data Leakage image and hacking case images are provided by the National Institute of Standards and Technology (NIST) [7]. Illegal Possession of images case image was contributed by Dr. Golden G. Richard III, and was originally used in the DFRWS 2005 RODEO Challenge [8]. Pixel 3 smartphone flash memory image is created by Joshua Hickman and hosted by digitalcorpora [9]. DJI Drone disk images are created by VTO Labs [10]. The case study covers GPS investigation and cached images retrieval.

B. Comprehensive labs

A comprehensive case study is essential for students to experience all phases of digital forensics. Table III shows the P2P data leakage case study with 11 labs. The case study mainly covers one device-specific KU and four domain-specific KUs. Topics covered by the case study are ranged from the lab environment setting up, disk image acquisition, disk media analysis, to the reconstruction of the timeline of the crime scene. Note that one topic may cover multiple domain-specific KUs, for example, investigating uTorrent log files may cover Network Forensics, Software Forensics, and even

TABLE II
REAL-WORLD CRIME CASES USED FOR DEVELOPING LAB MATERIALS

ID	Case Name	Description	Labs
1	NIST Data Leakage (Windows XP)	An image involving intellectual property theft	14
2	P2P Data Leakage (Windows 10)	Intellectual property theft involving P2P communications	11
3	Illegal Possession of Images	Illegal possession of Rhino images (Networking forensics)	6
4	Email Harassment	A harassment email was sent by a student to a faculty member	3
5	Illegal File Transferring	A memory image contains illegal file transferring from a server to a USB	3
6	Hacking Case	A hacker intercepted wireless network traffic	3
7	Pixel 3 Phone (Android 10)	Usages of Pixel 3 phone, including system and apps	3
8	iPhone (iOS 13)	Usages of iPhone, including system and apps	3
9	DJI Drone	Recover pictures and GPS trace from a DJI controller	3

Operating System Forensics. We only list the most important ones.

TABLE III
THE MAPPING BETWEEN DFAG-DRIVEN KUS AND TOPICS IN P2P DATA LEAKAGE CASE STUDY

Lab ID	Device-Specific KUs	Domain-Specific KUs	Topic Covered
1	Comp. Forensics	OS Forensics	Lab Environment Setting Up
2	Comp. Forensics	OS Forensics	Disk Image and Partitions
3	Comp. Forensics	OS Forensics	Windows Registry
4	Comp. Forensics	OS Forensics	MFT Timeline
5	Comp. Forensics	OS Forensics	USN Journal Timeline
6	Comp. Forensics	Network Forensics	uTorrent Log Files
7	Comp. Forensics	Media Forensics	File Signature
8	Comp. Forensics	Media Forensics	Emails
9	Comp. Forensics	Software Forensics	Web History
10	Comp. Forensics	Software Forensics	Website Analysis
11	Comp. Forensics	Media Forensics	Timeline (Summary)

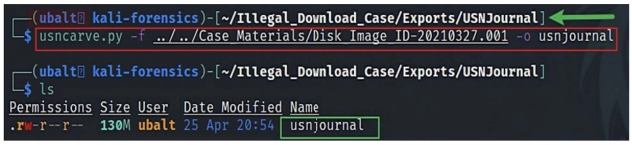
C. Problem-driven labs

We have utilized a problem-driven approach to design labs. The problem-driven approach allows students to provoke reflection, mobilize attention, and promote targeted and context-sensitive engagement when conducting case studies. For example, the Update Sequence Number (USN) Journaling is an important feature of Windows and the USN journal is an artifact of the Windows operating system. USN journal can be used for reconstructing users' behaviors. To cover the topic, the following questions are proposed based on the problem-driven approach:

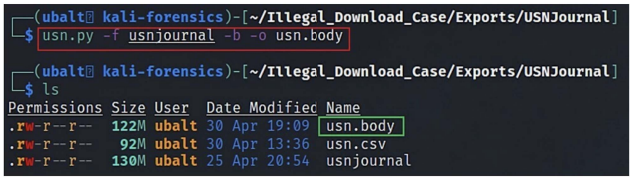
- What is the USN Journal?
- Where is a USN Journal located?
- How to get USN Journal?
- How to parse USN Journal?
- How to interpret parsed USN Journal?
- How to use *mactime* (a command-line tool from Sleuthkit) to generate a timeline of a specific file from the USN Journal records?

D. Self-paced learning

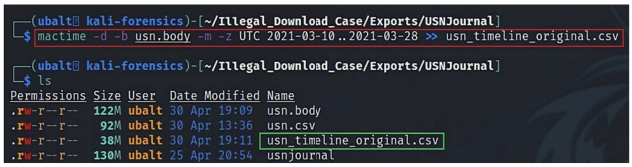
Self-paced learning enables students to customize their learning based on their diverse backgrounds, so that their individual study needs are fulfilled to the utmost. For each question proposed based on the problem-driven approach, solutions are provided with a detailed explanation, including theory, tools and commands used, command instructions, and screenshots of solutions. Figure 2 shows visualized instructions to (1) carve USN Journal from a raw image and save it to a text file; (2) parse carved USN Journal and save it to another text file; (3) generate timelines of files from the parsed USN Journal; (4) search timelines of files of interest, i.e., transferred audio files. Visualized instructions provides repeatable and intuitive solutions to help students with diverse backgrounds learn at their own comfortable pace.



Visualized instructions to carve USN Journal.



Visualized instructions to parse USN Journal.



Visualized instructions to generate timelines of files.

Visualized instructions to show one timeline of interest.

Fig. 2. Visualized instructions.

E. An integrated open-source lab environment

All digital forensics labs are hands-on intensive and rely on many open-source tools. Installing these tools is time-consuming and error-prone. There are a few well-known

Linux-based digital forensics platforms, such as SIFT[2] and CAINE[3]. Instead of using existing platforms, we have built our digital forensics platform based on Kali[4]. This is because Kali is a widely used platform in cybersecurity education. It is more practical for students and faculty to use Kali than SIFT and CAINE. We have designed two different approaches for students to set up the lab environment: (1) importing our customized Kali Virtual Machine (VM) image with all the necessary open-source tools pre-installed; or, (2) installing open-source tools in Kali by themselves using the shell script we have developed. The customized VM and shell script are also available at GitHub repository. There are three types of investigating tools included in the VM:

1) Tools provided by Ubuntu: commonly used tools include xxd, find, grep, date, etc.
2) Tools included in Kali: They include TSK and Autopsy, Wireshark, PhotoRec. Note that TSK itself includes a list of commands, such as fls, icat, etc.
3) Tools need to be re-compiled and installed: They include regripper, usn parser, AnalyzeMFT, etc. These tools are often installed via apt install commands or need to be downloaded from GitHub and installed manually.

Figure 3 shows the execution of the customized shell script to install the third type of tools. Table IV shows a tracing table that maps among labs, the artifacts to investigate, and Linux tools and commands used to investigate artifacts in the P2P data leakage case study. The table provides a basis for evaluating the coverage of lab topics, artifacts, and tools.

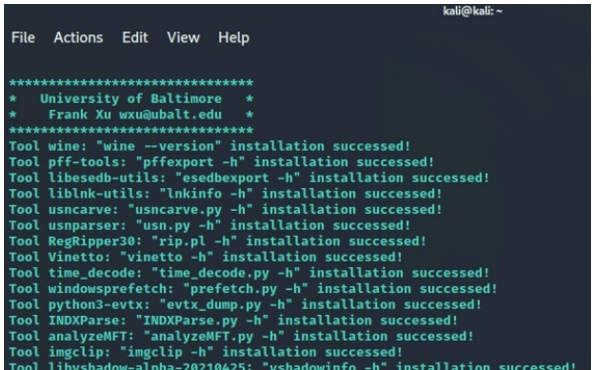


Fig. 3. The third type tools installed using the customized shell script.

V. Implementation of Instructional Materials

We have developed the instructional materials and shared them on GitHub. Figure 4 provides an overview for contents of the materials. This open source repository has drawn lots of attention since being published in Summer 2021 with nearly 600 stars as shown in Figure 5 and 22k views based on the visitor tracking website[5]. In addition, the GitHub traffic history[6] shows that our instructional materials have been referred by many social media and cybersecurity websites, including Twitter, Facebook, LinkedIn, kitploit.com, onehack.us, etc.

[2]https://www.sans.org/tools/sift-workstation
[3]https://www.caine-live.net
[4]https://www.kali.org
[5]https://trackgit.com/

TABLE IV
Mappings among labs, artifacts, and Linux tools and commands in P2P data leakage case study

Lab	Artifacts	Linux Tools and Commands Used
1		VirtualBox, Kali, Windows 10 cd, mkdir, pwd, wget, nano, bash, sudo, alias, ls
2	.dd (a disk image)	Md5deep, sha1deep, fdisk, mmls, parted, fsstat, fls, Regripper, hivexsh
3	.dd, NTUSER.DAT, System, Software, Sam uTorrent-relatec files Browsers	losetup, regripper, hivexsh, grep, head, fls
4	.dd, $MFT	icat, AnalyzeMFT, mactime, Libreoffice Calc
5	$UsnJrn:$J,	grep, USN Journal Parser, USN Record Carver, fls, Log2timeline, tsk_gettimes
6	NTUSER.DAT	regripper, icat, torrent-file-editor
7	torrent-related files	hexedit, icat, MD5deep, SHA1deep, Hashdeep, MD5sum, SHA1sum, exiftool, strings
8	Software, Mozilla Thunderbird.lnk, office365.com/INBOX	lnkinfo, regripper, icat ,mutt
9	Edge/.../.../History, WebCacheV01.dat,	icat, sqlites
9	dfir-projects.boards.net/, Sample-1.mp3.torrent	torrent-file-editor

VI. Related Works

Many efforts have been made on designing and developing shareable digital forensics instructional materials for digital forensics communities [11] [12] [13] [14] [15] [16] [17]. These materials often have one or multiple weaknesses, including insufficient KU and topic coverage, the lack of comprehensive case studies, publicly unavailable labs. Johnson et al. [17] developed instruction questions to help students understand the targeted digital forensics concepts, which is similar to our problem-driven approach. However, these questions are derived from lectures, not hands-on labs. There are a few shared digital forensics repositories on GitHub [18] [19]. These repositories mainly use labs from other textbooks and labs have similar weakness we have mentioned earlier. The repository of Costello [20] contains a serial of lab assignments for the IoT Digital Forensics Course. IoT devices included in their lab assignments are Alexa echo, smartwatch, and Fitbit. Each lab assignment comes with necessary logic files extracted from IoT flash memory. However, flash memory images and solutions are not available.

VII. Conclusion and Future Work

To promote digital forensic education, we have systematically developed digital forensics instructional materials based on real-world cybercrime cases. These active learning-oriented

[6]https://github.com/frankwxu/digital-forensics-lab/graphs/traffic

Table of Contents (add iOS investigation on 4/21/2022)

- Basic Computer Skills for Digital Forensics
  - Number Systems
  - PC Introduction
  - Windows Command Line Tutorial
  - Linux Command Line Tutorial
  - Advanced Linux Command Line Tutorial
- Computer and Digital Forensics (updated on Oct. 2021)
  - Introduction to Digital Forensics
  - Sleuth Kit Tutorial
  - USB Image Acquisition
  - Evidence Search - A Pattern Match Game updated on May 2022
  - Evidence Search - File Metadata
  - Data Carving
  - Steganography
  - Forensic Report Template
- Computer Forensics Case Study
  - Investigating NIST Data Leakage (Windows XP)
  - Investigating P2P Data Leakage (Windows 10)
  - Investigating Illegal Possession of Images ("Networking forensics")
  - Investigating Email Harassment
  - Investigating Illegal File Transferring (Memory Forensics)
  - Investigating Hacking Case
- Mobile/IoT Forensics Case Study
  - Investigating Android 10 (added on 10/24/2021)
  - Investigating iPhone iOS 13 (added on 4/21/2022)
  - Investigating Drone (add on 12/07/2021)
- Forensic Intelligence Repository
  - Email forensics
  - Illegal Possession of Images

Fig. 4. Table of contents of instructional materials.

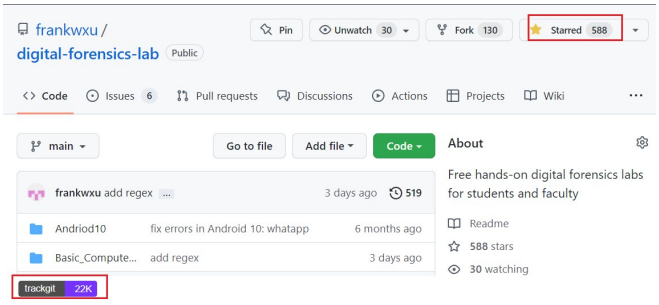


Fig. 5. The shared instructional materials on GitHub.

materials can assist students to master necessary digital forensics skills. For future work, we plan to add more real-world case studies to our repository. To keep improving our work, we would like to conduct a formal evaluation and collect feedback from students and faculty. We also plan to formalize evidence to better share digital forensic evidence with communities.

ACKNOWLEDGMENT

We thank Malcolm Hayward, Richard Wheeles, Harleen Kaur, and Danny Ferreira for assistance with developing three case studies, including P2P Data Leakage, Hacking, Pixel 3 smartphone, and iOS usages. This work is supported in part by the U.S. Department of Justice under 2019-DF-BX-K001 and the Department of Homeland Security Science and Technology Directorate Office of University Programs (2021-2022). Deng is supported as a Jess and Mildred Fisher Endowed Professor of Computer and Information Sciences from the Fisher College of Science and Mathematics at Towson University. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the agency.

REFERENCES

[1] B. Hooke, "On the horizon: Tech trends impacting law enforcement investigations." https://www.police1.com/police-products/investigation/computer-digital-forensics/articles/on-the-horizon-tech-trends-impacting-law-enforcement-investigations-y3qMdL63eQjUTOoP/, 2018. Accessed: 2022-01-01.

[2] www.bls.gov, "Information security analysts." https://www.bls.gov/ooh/computer-and-information-technology/information-security-analysts.htm/, 2021. Accessed: 2022-01-01.

[3] DFRWS, "A road map for digital forensic research," in *Proceedings of The Digital Forensic Research Conference*, 2001.

[4] M. McCormick, "Scientific evidence: Defining a new approach to admissibility," *Iowa L. Rev.*, vol. 67, p. 879, 1981.

[5] M. Bishop, D. Burley, and L. A. Futcher, *Cybersecurity Curricular Guidelines*, pp. 158–180. Cybersecurity Education for Awareness and Compliance, Hershey, PA, USA: IGI Global, 2019.

[6] C. C. Bonwell and J. A. Eison, *Active Learning: Creating Excitement in the Classroom. 1991 ASHE-ERIC Higher Education Reports.* ERIC, 1991.

[7] NIST, "Data leakage case." https://cfreds-archive.nist.gov/data$_l eakage_c ase/data$ – $leakage$ – $case.html$, 2018. $Accessed$: 2022 – 01 – 01.

[8] G. Richard, "Rhino hunt." https://cfreds-archive.nist.gov/dfrws/Rhino$_H unt.html$, 2005. $Accessed$: 2022 – 01 – 02.

[9] J. Hickman, "Android 10 images." https://digitalcorpora.org/corpora/cell-phones/android-10, 2020. Accessed: 2022-01-02.

[10] VTOLabs, "Drone forensics." https://www.vtolabs.com/drone-forensics, 2021. Accessed: 2022-01-02.

[11] T. M. Vidas, D. A. Branch, and A. Nicoll, "Stealing lab support in digital forensics education," in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, pp. 482–482, IEEE, 2008.

[12] K. R. Lawrence and H. Chi, "Framework for the design of web-based learning for digital forensics labs," in *Proceedings of the 47th Annual Southeast Regional Conference*, pp. 1–4, 2009.

[13] H. Chi, F. Dix-Richardson, and D. Evans, "Designing a computer forensics concentration for cross-disciplinary undergraduate students," in *2010 Information Security Curriculum Development Conference*, pp. 52–57, 2010.

[14] M. Simmons and H. Chi, "Designing and implementing cloud-based digital forensics hands-on labs," in *Proceedings of the 2012 Information Security Curriculum Development Conference*, pp. 69–74, 2012.

[15] M. M. Parvez, S. A. Hossain, and S. M. R. Ali, "Design and implementation of low cost digital forensic laboratory for university," in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 1524–1528, IEEE, 2017.

[16] X. Wang, Y. Bai, and B. S. Goda, "Project design and implementation for digital forensics education," *Proceedings of the 20th Annual SIG Conference on Information Technology Education*, p. 33–38, 2019.

[17] W. Johnson, I. Ahmed, V. Roussev, and C. B. Lee, "Peer instruction for digital forensics," in *2017 USENIX Workshop on Advances in Security Education (ASE 17)*, (Vancouver, BC), USENIX Association, Aug. 2017.

[18] K. ANamin, "Instructional materials for the digital forensics course." http://github.com/asiamina/A-Course-on-Digital-Forensics, 2021. Accessed: 2022-01-01.

[19] S. Parasram, "Digital forensics with kali linux." https://github.com/PacktPublishing/Digital-Forensics-with-Kali-Linux, 2021. Accessed: 2022-01-01.

[20] J. Costello, "Iot-digital-forensics-course." https://github.com/RJC497/IoT-Digital-Forensics-Course, 2020. Accessed: 2022-01-01.