User Profiling Attack Using Windows Registry Data

Edward L. Amoruso
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816 US
eamoruso@knights.ucf.edu

Richard Leinecker
Department of Computer Science
University of Central Florida
Orlando, FL 32816 US
Richard.Leinecker@ucf.edu

Cliff C. Zou
Department of Computer Science
University of Central Florida
Orlando, FL 32816 US
changchun.zou@ucf.edu

***Abstract*—The Windows registry stores a glut of information containing settings and data utilized by the Microsoft operating system (OS) and other applications. For example, information such as user credentials, installed programs, recently used applications and documents, accessed resources such as local, remote, and removable devices can all be found in this database. More revealingly, the registry also has time and date stamps that can help build a timeline of user activities. The Windows registry can be easily queried by either malicious or benign applications. This is possible through the Windows Application Program Interface (API) and other OS built-in utilities. In this paper, we develop and demonstrate a program able to collect and infer a user's rich activities by accessing the Windows registry alone. This information, also referred to as the user's digital footprint, can be used to devise an exploit or create a privacy threat. Our custom developed application will demonstrate how a user's digital footprint can be acquired by a malicious application from a Windows registry, without alerting security software. In addition, this information can be exported to a set of comma delimited files, making it easy to import them into other analysis applications.**

Keywords—Windows registry, user digital footprint, registry analysis, event timelines, event frequency, inferable data

I. Introduction

The Windows registry, which will be referred to as the registry for the remaining of this paper, holds an overabundance of information. The registry is used by various applications, including the Windows operating system (OS) with the intended use to keep configuration data. This data is used for systemwide and per-user settings, but a fair amount of it contains system and user identifiable information. For example, items such as:

- Type of computer and hardware
- Windows operating system and version number
- Computer system's current time zone
- Applications installed and available to users
- History of recently accessed files
- Available network resources
- Network settings, such as IP address and gateway

Furthermore, many entries in the registry contain date and time information, making it possible to infer many user activities. Even non-cyber activities can be inferred such as user's working performance (e.g., is she or he actively working on company assigned tasks), user's job type (e.g., is she an accountant or programmer based on her modified file names). Also, other insight can be deduced such as typical workday hours.

When a user executes an application, he or she will generally leave behind some type of data in the registry. For example, if opening a folder, the date and time and source location will be recorded. Having date and time information can help create a timeline of activities performed by that user. In other scenarios, such as opening a document, the action will also create an entry of the file name in the registry. This information will be referred to as the user's digital footprint. Subsequently, the term digital footprint has a broad definition, in our paper, we will use the expression to identify user and system information acquired throughout the registry for building a user profile.

The registry can be accessed by various techniques and applications. For example, an attacker or common user can run the Window's built-in command-line utility called "reg.exe" to extract data from the registry. Similarly, a more powerful tool, the Window's PowerShell, a cross-platform task automation solution, can be used to access registry information [1]. Both mentioned programs evade security applications since they are included and available to the OS. According to Mandiant, the PowerShell is predominately used among malicious actors because it is a trusted environment allowing it to be less suspicious to identify for nefarious activities [2]. Alarmingly, as stated by RANGEFORCE, PowerShell has been deployed by top cybercriminals such as APT1, Duqu, and APT10 to gather critical intelligence to assist in sophisticated cyberattacks [3]. There are also several third-party applications available on the internet with various functionality to extract registry information [5] [6] [7]. Some of these programs are typically written and intended for digital forensic investigations. Others, used by penetration testers, such as PowerShell Empire and Covenant, can also exfiltrate registry information [8] [9].

Addressing security access is a pivotal consideration during the retrieval of data from the registry. Certain registry keys necessitate elevated permissions, typically in the form of administrative access, for successful retrieval. Moreover, the Windows Operating System incorporates a built-in security mechanism known as "User Account Control" (UAC). This feature operates by default and functions to mitigate the potential impact of malicious software. It achieves this by mandating user approval through prompts whenever a program

necessitates privileged system access. The primary intent behind this notification is to notify the user, thereby prompting a sense of wariness and preempting potential cyberattacks.

Nonetheless, our research underscores a remarkable finding: the acquisition of all requisite information was accomplished without setting off UAC prompts or demanding heightened privileges. This revelation bears notable significance, given that malevolent actors frequently exploit compromised accounts stemming from phishing endeavors. Typically, these ill-gained accounts lack administrative privileges, especially within environments that prioritize stringent security measures.

This paper focuses on the collection of comprehensive information from the system registry. This is achieved by utilizing both the operating system's native application programming interface (API) and a custom-developed program. The primary aim is to construct a detailed digital footprint of a user. Leveraging the native API allows us to streamline development by building upon an existing foundation, thereby reducing the amount of code and time required for development. Additionally, employing native APIs provides an added advantage of circumventing security applications such as Windows Defender. The objective of this research is to uncover a significant real-world threat. We achieve this by meticulously extracting ample registry data, all while operating within standard privileges. The outcome of this process results in a potentially hazardous vulnerability, posing risks to both privacy and security.

The contributions of this paper are:

- A detailed approach in gathering enough registry information to produce a user's digital footprint that can expose the user to privacy and security breaches.
- Provide a ready-to-use application capable of seamless execution, bypassing any potential conflicts with the operating system or security software to easily create the digital footprint of the user's account on the target Windows system.
- Conduct experiments using real-world registry data collected from multiple businesses, illustrating the extent to which a user's digital footprint can be derived solely from registry data.

The rest of this paper is organized as follows. Section II covers an overview of other related works and how our paper differs. In Section III, we introduce our proposed approach. Section IV we discuss our implementation. Section V will go over an evaluation based on several real-world samples. Then in Section VI, we discuss limitations and future work. Finally, in section VII we provide our conclusion.

II. Related Work

Presently, there are many solutions available to help acquire a user's information from his or her computer. Some are automated scripts or programs developed to extract various data [10] [11] [12], others are manual techniques using built-in OS applications [13] [14]. In our research, we aim to leverage a technique and data repository to easily acquire a user's digital footprint. With this in mind, we focus on a single source, Windows registry, for gathering a user's profile. Furthermore, we explore methods for which the information can be automatically gathered with no resistance from security software.

Throughout our assessment, we examined many tools and practices documented by educational and other entities. Most provided a process to acquire evidence from the registry for digital forensic analysis [13] [14]. Also, several resources offered techniques and cheat sheets to extract registry information manually [15] [16]. Other research use machine learning (ML) and artificial intelligence (AI) in creating a user's digital footprint, associated to performance analysis and anomalous activity detection [17].

While the previously mentioned approaches show promise, their primary limitation lies in neglecting the automated extraction of registry data to establish a user's digital footprint, thereby making them only feasible in theory but too complicated and time-consuming to be utilized by most people. Taking our research one step further, we unveil the actual vulnerability, highlighting how attackers can effortlessly amass a user's digital footprint from their computer by simply accessing registry data. This process is made even smoother by utilizing established Windows APIs, sidestepping potential obstacles posed by system constraints and security software.

III. Our Proposed Approach

In this section we describe how our custom application extracts registry data, without alerting security software, to construct a user's digital footprint. Our goal is to rapidly extract the necessary values (e.g., usernames, applications, hardware, accessed resources, time and date stamps) through predefined registry key locations. This eliminates the need to traverse the entire registry, improving performance and execution time. For our prototype, all results needed for a user's digital footprint are saved to a locally created folder. Each file contains information essential for either compromising or inferring the following:

- User's computer environment, whether it is virtual or bare metal machine, and a high-performance device.
- User's computer usage patterns, such as access to most recently used (MRU) files (e.g., Word, Excel, PowerPoint).
- Repetitions on folders and devices accessed by the user.
- A timeline of a user's cyber activities, including program names and access time, file names and access/modification time, etc. Such a comprehensive timeline of cyber activities would enable us to infer user's private and non-cyber activities, such as work hours, lunch breaks, job type (e.g., an accountant or programmer) or time-taken-off by the user.
- Sensitive data, files identified by their corresponding names and locations (e.g., payroll, prototype design, sale commissions).
- Additional data extracted, like the user's device type, computer name, and installed software, can be utilized to infer the individual's occupational category. To illustrate, consider a user managing a system referred to

as "sys-CAD-cpu01," which includes drafting applications; this scenario might indicate that the user holds a profession related to computer-aided design (CAD).

- The security posture of the user's system can be assessed by categorizing both startup and installed programs. This classification could potentially aid an attacker in recognizing security applications, allowing them to either bypass or deactivate security software.

A. Digital Footprint Privacy

In this paper, we are not concerned about discovering the user's identity, instead we compile a profile to understand his or her responsibilities and activities, as they relate closely to privacy. The captured behaviors in turn will help assemble a timeline, exposing a user's cyber and non-cyber routines and processes. This also includes other information such as frequently accessed resources (e.g., hard drive, network, external devices), to help infer the user's responsibilities.

B. Digital Footprint Security

We establish that the digital traces generated through registry data have the potential to compromise the security of the user's system and the privacy of the user. The information collected includes the locations of files, usernames, hardware specifications, and installed software. This type of information exposure can make the user susceptible to additional attacks or vulnerabilities. Consider the following scenario: having access to information about the programs within a system could potentially become a weak point for malicious actors to exploit. This entails identifying vulnerabilities within these applications, which could then be used to infiltrate the system even more profoundly.

To illustrate further, let's take an example involving intellectual property (IP). A criminal could strategically focus on stealing a user's documents if they possess pieces of information of the IP. Lastly, disclosing hardware specifics to a malicious actor could reveal whether the device is powerful enough to execute a desired assault, such as a distributed denial of service attack (DDoS).

C. Retrieving Registry Key Data

Numerous methods exist for extracting data from the registry. For someone new to computer usage, the integrated Windows command-line tools, such as reg.exe, can be employed to extract either individual or multiple values. This method necessitates that the user is familiar with each key necessary for constructing a user's profile. Consequently, they must undertake the laborious task of manually gathering and parsing data to piece together the user's digital footprint. In essence, current approaches demand that an attacker possesses extensive knowledge of the registry and engages in intensive operations to profile users on a targeted Windows machine.

In our approach, using C++ programming language and Windows built-in APIs, we automate the parsing of specific registry key data and assemble them into meaningful information. This is possible with extensive examination and research of the registry; we collect the requisites for building a user's digital footprint. Other insight on the registry was obtained from several digital forensic cheat sheets [15] [16].

The purpose of our application is to efficiently arrange extracted registry details, facilitating coherent data access and identification. Once our specialized program processes the information, it generates a designated folder containing appropriately categorized files based on their types. In our initial demonstration, we opted to store these files directly on the targeted device. However, it's worth noting that this functionality could readily be adapted to enable uploading to a website or an internet server if desired.

D. Storing Captured Registry Data

In our prototype, we generate six distinct files, each encompassing particular pieces of information that we've collected from the registry. To enable seamless integration into external applications such as Excel, Google Sheets, and Calc, we structure the data in each file using the comma-separated values (CSV) format. This formatting choice offers numerous advantages, primarily facilitating the transfer of data from one application to another. One particularly noteworthy benefit, as highlighted in subsequent sections of this paper, is its capacity to facilitate the creation of visual representations like timeline graphs through tools like Excel.

Of the six files, three of the files contain different Office applications' information to help build a list of most recently used (MRU) documents, spreadsheets, and presentations by the user. The fourth file is created to store general information such as system hardware, network settings, installed software, and user credentials. Finally, the last two files store date and time stamps of user activities and all user accounts on the target system.

E. Avoiding Security Software

To evade security software, we use built-in Windows APIs to retrieve our data from registry keys. These registry keys require no additional privileges for admittance, making it seamless for access by any regular user account. Equally, using native APIs, we can avoid system monitoring software, a concept referred to as living off the land. According to [18], this is considered a tactic used by malicious actors to evade detection and blend in as normal activities. Both methods allow us to gather information unnoticed and efficiently from the registry.

IV. Implementation

The Windows registry is a specialized database that the Windows operating system maintains. For instance, if you install Microsoft Office, the registry contains data paths, plugins, language information, and anything else the program needs to function with. While the amount of information that an application such as Office saves to the registry is large, most other applications at the same time also install their own configuration data. Interestingly, the registry has become a "De Facto" for application developers to use to store their program settings. To illustrate the importance of registry data, if the registry gets corrupted, the Windows operating system will not function correctly.

This paper is intended to show how user information can be harvested to create a user's digital footprint based on registry only. A great deal of useful information can be found in the registry. For the purposes of this paper, we will categorize what we examine as basic user data, general hardware and software information, the file usage history for users, and recent document handling. To research this, we create an application that examines the Windows registry and extracts the relevant artifacts. In the next several sections these will be explained. Our developed application code is publicly available on GitHub at "https://github.com/eamoruso/UserProfileAttack" link.

When our custom-developed application runs, a folder named **mm-dd-yyyy** is created, which is named with the current date. For example, if we run our program on February 11, 2023, the folder will be called "02-11-2023" within the same directory as our executing application. Report CSV files are saved into this folder.

A. User's Basic Information

The first step to gather basic user information is to enumerate the registry key called "**ProfileList"** located in "**SOFTWARE\Microsoft\Windows NT\CurrentVersion\"**. Each key represents a user's security ID. The data contained within each key has relevant information such as ProfileImagePath, Flags, and Security Identifier (SID). Together, this information can positively identify a user to the operating system. For the purposes of this paper, the ProfileImagePath and SID are saved to a CSV within the newly created folder named Basic.CSV.

One other important detail that can be gathered from here is a list of software that each user operates. This can narrow down what program is used by each user and when he or she installed it. In many cases we can infer with high confidence what type of job the user has in the business. For example, if several accounting applications (e.g., QuickBooks, Sage, NetSuite) are listed, one can infer that the user is responsible for bookkeeping or a finance related job function. Fig. 1 shows the contents of an example Basic.CSV file.

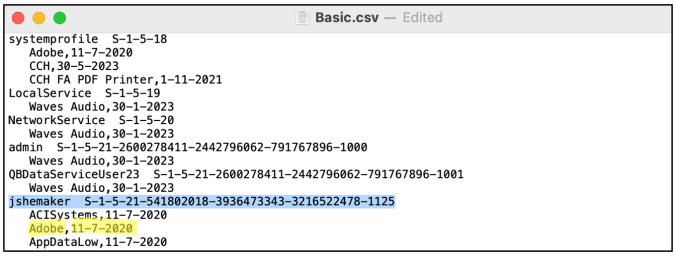


Fig. 1. Basic User Information (ProfileList) file contains all available users on the system. For example, "jshemaker" is the username followed by the Securtiy Identifier (SID), which starts with "S" and ends with numbers. The next few lines show programs installed and installation dates by this user.

B. Identifying and Retrieving Personal Data

The next resource that can be dug to extract personal information is the list of the documents, spreadsheets, and presentations that a targeted user has edited. For this paper we will gather all documents, spreadsheets, and presentations created or modified with Microsoft Word, Excel, and PowerPoint. Although, there are other such programs (e.g., LibreOffice, Apache OpenOffice) capable of also creating these types of files, we only focus on Microsoft Office in this paper, which can be easily expanded to cover other types of documents and programs in the future. There are several factors that help solidify our choice to use Microsoft. Office applications are predominant, according to Enlyft, Microsoft Office has 45.46% Market Share [19]. Also, all the private entities willing to participate in our testing only used the Office suite. Finally, Office applications store all their most recently used (MRU) files in the registry, shown in Fig. 2. Other applications, such as OpenOffice, store few settings, but nothing related to the opening of any documents, spreadsheets, or presentations.

It is important to note that our custom designed application does not open all users' MRUs, only the targeted user's MRU. In other words, the application only extracts information from the user that is currently logged into the system. The target user's MRU consists of a list of file names, each can be opened and read to gain significant information about the active user. These files could also be downloaded, possibly with an automated retrieval program if the program has this target user's account access.

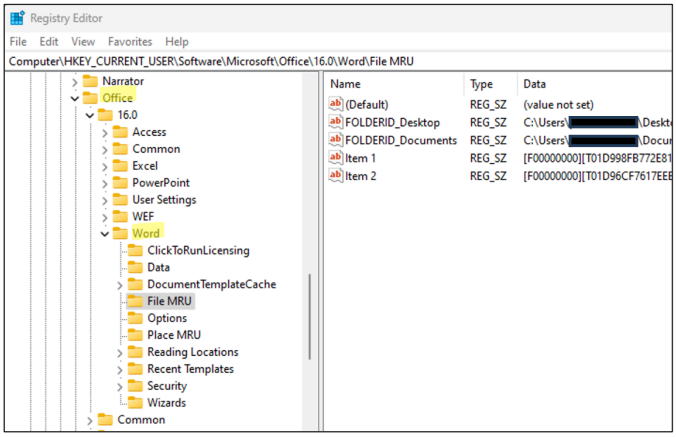


Fig. 2. Excerpt of the Microsoft Office Word File MRU containing most recently used document file(s).

In gathering the user's access history to all Word and Excel files, we start with the most recently used (MRU) entries found in the registry. The key is **SOFTWARE\Microsoft\Office** followed by the version of Office that is installed. For our development system, we use the most recent available version **16.0** (e.g., *SOFTWARE\Microsoft\Office\16.0***)** which covers Office 365, 2019, and 2016 [20]. Each application's subkey is shown in Table I. Within that key are subkeys for each of the Office applications. For this paper we limit the search to Word, Excel and PowerPoint.

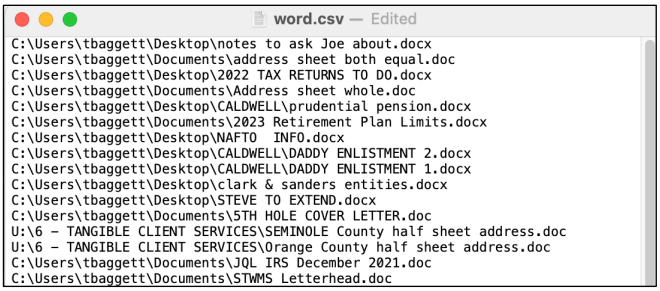
Within each of the Office and Word subkeys are further subkeys organized into two categories: files and places; represented by **\User MRU\File MRU** and **\User MRU\Place**

MRU. The files' subkey lists the documents that have been edited while the places' subkey lists the locations of those files.

TABLE I. OFFICE VERSION 2016, 2019, 365 MRU ENTRIES

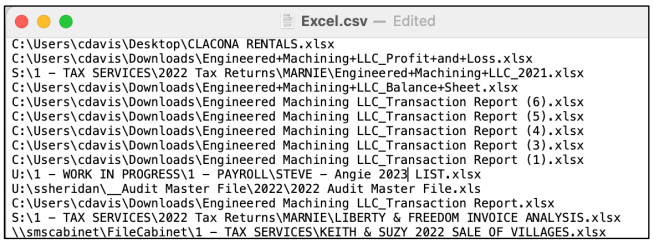
Program	Registry Subkey Location
Word	HKCU\SOFTWARE\Microsoft\Office\16.0\Word
Excel	HKCU\SOFTWARE\Microsoft\Office\16.0\Excel
Access	HKCU\SOFTWARE\Microsoft\Office\16.0\Access
Outlook	HKCU\SOFTWARE\Microsoft\Office\16.0\Outlook
PowerPoint	HKCU\SOFTWARE\Microsoft\Office\16.0\PowerPoint

The results are then saved into three files named Word.CSV, Excel.CSV, and Powerpoint.CSV. To demonstrate this, Fig. 3, Fig.4, and Fig. 5 show the Word, Excel, and PowerPoint CSV files with redacted, modified private information since they are extracted from real business computers.


word.csv — Edited

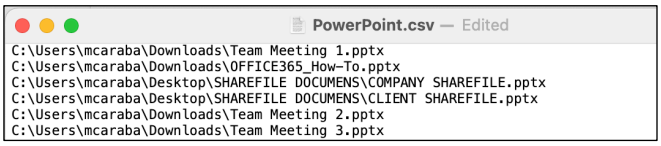
```
C:\Users\tbaggett\Desktop\notes to ask Joe about.docx
C:\Users\tbaggett\Documents\address sheet both equal.doc
C:\Users\tbaggett\Desktop\2022 TAX RETURNS TO DO.docx
C:\Users\tbaggett\Documents\Address sheet whole.doc
C:\Users\tbaggett\Desktop\CALDWELL\prudential pension.docx
C:\Users\tbaggett\Documents\2023 Retirement Plan Limits.docx
C:\Users\tbaggett\Desktop\NAFTO  INFO.docx
C:\Users\tbaggett\Desktop\CALDWELL\DADDY ENLISTMENT 2.docx
C:\Users\tbaggett\Desktop\CALDWELL\DADDY ENLISTMENT 1.docx
C:\Users\tbaggett\Desktop\clark & sanders entities.docx
C:\Users\tbaggett\Desktop\STEVE TO EXTEND.docx
C:\Users\tbaggett\Documents\5TH HOLE COVER LETTER.doc
U:\6 - TANGIBLE CLIENT SERVICES\SEMINOLE County half sheet address.doc
U:\6 - TANGIBLE CLIENT SERVICES\Orange County half sheet address.doc
C:\Users\tbaggett\Documents\JQL IRS December 2021.doc
C:\Users\tbaggett\Documents\STWMS Letterhead.doc
```

Fig. 3. The Word CSV File shows a list of all most recently used Word file(s) accessed by the target user.


Excel.csv — Edited

```
C:\Users\cdavis\Desktop\CLACONA RENTALS.xlsx
C:\Users\cdavis\Downloads\Engineered+Machining+LLC_Profit+and+Loss.xlsx
S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\Engineered+Machining+LLC_2021.xlsx
C:\Users\cdavis\Downloads\Engineered+Machining+LLC_Balance+Sheet.xlsx
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report (6).xlsx
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report (5).xlsx
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report (4).xlsx
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report (3).xlsx
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report (1).xlsx
U:\1 - WORK IN PROGRESS\1 - PAYROLL\STEVE - Angie 2023 LIST.xlsx
U:\sheridan\__Audit Master File\2022\2022 Audit Master File.xls
C:\Users\cdavis\Downloads\Engineered Machining LLC_Transaction Report.xlsx
S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\LIBERTY & FREEDOM INVOICE ANALYSIS.xlsx
\\smscabinet\FileCabinet\1 - TAX SERVICES\KEITH & SUZY 2022 SALE OF VILLAGES.xlsx
```

Fig. 4. The Excel CSV File shows a list of all the most recently used Excel spreadsheet file(s) by the target user.


PowerPoint.csv — Edited

```
C:\Users\mcaraba\Downloads\Team Meeting 1.pptx
C:\Users\mcaraba\Downloads\OFFICE365_How-To.pptx
C:\Users\mcaraba\Desktop\SHAREFILE DOCUMENS\COMPANY SHAREFILE.pptx
C:\Users\mcaraba\Desktop\SHAREFILE DOCUMENS\CLIENT SHAREFILE.pptx
C:\Users\mcaraba\Downloads\Team Meeting 2.pptx
C:\Users\mcaraba\Downloads\Team Meeting 3.pptx
```

Fig. 5. The PowerPoint CSV File shows a list of all recently used PowerPoint presentation file(s) accessed by the target user.

Note that Access and Outlook all have their own MRU sections. The software written for this paper can be easily amended to query these. For the program as it exists, this method is called three times with "Word," "PowerPoint," and "Excel" parameters. We can easily add "Access" and "Outlook" parameters to create additional extracted information files for these applications.

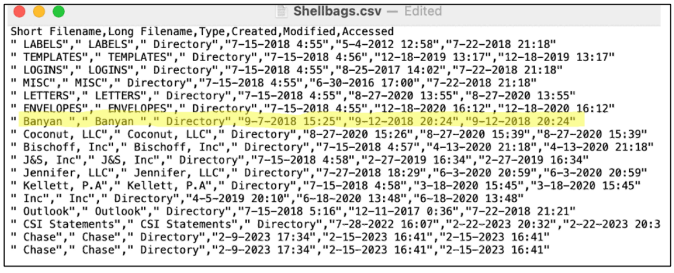
C. *Building Timeline of Activities*

There are two registry subkeys that contain information called Shellbags. These subkeys record all folder operations that a user performs [13]. For instance, if someone resizes a window, that action will be recorded as Shellbag information. If a folder is opened by an application, then that application name will be recorded. The valuable information contained in Shellbags is that a timeline for user navigation and usage can be tracked. The different types of information from the two registry subkeys that contain Shellbag data are both "**USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\BagMRU**" and "**USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\Bags,**" described and shown in Table II.

There is an immense amount of information in these Shellbags. To produce a manageable subset for this paper, we simply examine a few months of data, since Windows may store such data for several years. An example of this file can be seen in Fig. 6.

TABLE II. SHELLBAG DATA

Type of Information	Registry Subkey Location
Stores folder names and records the folder paths	USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\BagMRU
Stores the view preferences such as the window size, location and view mode	USRCLASS.DAT\Local Settings\Software\Microsoft\Windows\Shell\Bags


Shellbags.csv — Edited

```
Short Filename,Long Filename,Type,Created,Modified,Accessed
" LABELS"," LABELS"," Directory","7-15-2018 4:55","5-4-2012 12:58","7-22-2018 21:18"
" TEMPLATES"," TEMPLATES"," Directory","7-15-2018 4:56","12-18-2019 13:17","12-18-2019 13:17"
" LOGINS"," LOGINS"," Directory","7-15-2018 4:55","8-25-2017 14:02","7-22-2018 21:18"
" MISC"," MISC"," Directory","7-15-2018 4:55","6-30-2016 17:00","7-22-2018 21:18"
" LETTERS"," LETTERS"," Directory","7-15-2018 4:55","8-27-2020 13:55","8-27-2020 13:55"
" ENVELOPES"," ENVELOPES"," Directory","7-15-2018 4:55","12-18-2020 16:12","12-18-2020 16:12"
" Banyan "," Banyan "," Directory","9-7-2018 15:25","9-12-2018 20:24","9-12-2018 20:24"
" Coconut, LLC"," Coconut, LLC"," Directory","8-27-2020 15:26","8-27-2020 15:39","8-27-2020 15:39"
" Bischoff, Inc"," Bischoff, Inc"," Directory","7-15-2018 4:57","4-13-2020 21:18","4-13-2020 21:18"
" J&S, Inc"," J&S, Inc"," Directory","7-15-2018 4:58","2-27-2019 16:34","2-27-2019 16:34"
" Jennifer, LLC"," Jennifer, LLC"," Directory","7-27-2018 18:29","6-3-2020 20:59","6-3-2020 20:59"
" Kellett, P.A"," Kellett, P.A"," Directory","7-15-2018 4:58","3-18-2020 15:45","3-18-2020 15:45"
" Inc"," Inc"," Directory","4-5-2019 20:10","6-18-2020 13:48","6-18-2020 13:48"
" Outlook"," Outlook"," Directory","7-15-2018 5:16","12-11-2017 0:36","7-22-2018 21:21"
" CSI Statements"," CSI Statements"," Directory","7-28-2022 16:07","2-22-2023 20:32","2-22-2023 20:3
" Chase"," Chase"," Directory","2-9-2023 17:34","2-15-2023 16:41","2-15-2023 16:41"
" Chase"," Chase"," Directory","2-9-2023 17:34","2-15-2023 16:41","2-15-2023 16:41"
```

Fig. 6. The Shellbags CSV File shows all folders and files accessed by a user with date and time stamps, for example, "Banyan" was last accessed December 9th, 2018 at 20:24 UTC. Also, Banyan is identified as a folder name.

D. *Extracting General Information*

Finally, we can also extract general information containing hardware, software, and network settings from the registry. To obtain this information, we create two separate functions, to facilitate extracting from the different registry subkeys. For example, some subkeys have specific value name/data, other subkeys just have all value name/data in the same subkey.

During the process of collecting all the data, our application stores this information to a text (TXT) file named GeneralInfo.TXT. An example of this file is shown in Fig. 7.

Information such as system's manufacturer, product name, and basic input/output system (BIOS) details can all be found under the "**HARDWARE\DESCRIPTION\System\"** subkey in the registry**.** An interesting fact about this subkey is that the

OS creates it during system boot and is stored entirely in memory. Although this subkey also contains a list of processors, for coding simplicity, we use the value data from "PROCESSOR_IDENTIFIER" under "**SYSTEM\CurrentControlSet\Control\SessionManager\Environment"** subkey. The processor count and architecture are also recorded under this subkey.

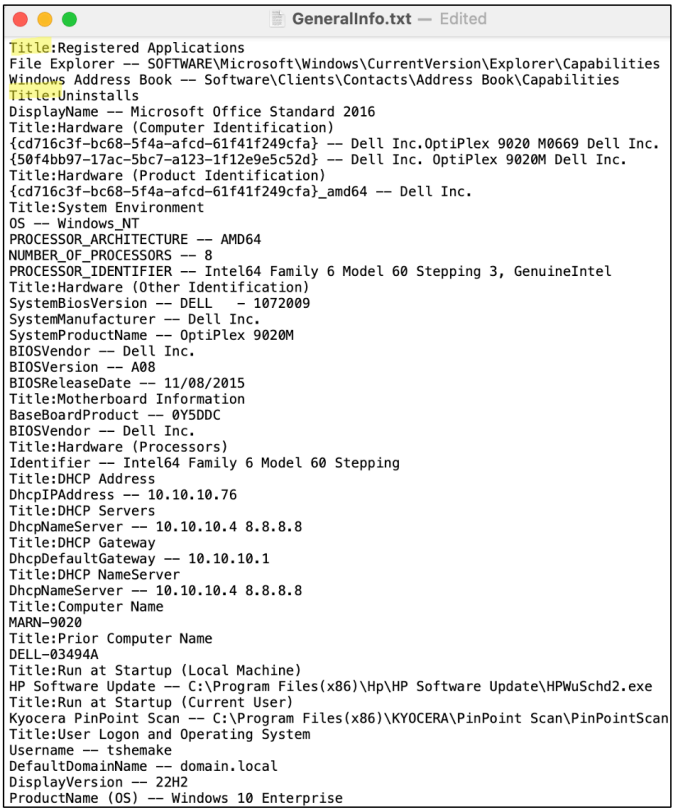


Fig. 7. General Information (Hardware, Network & Software settings), this is an excerpt with many redundant values removed. The highlighted "Title" serves as a seperator for each retievied category of information.

The remaining registry data obtained for our general information file is:

- Operating System and version information
- Username login credential
- User registered applications
- All Installed programs and patches (uninstall)
- Internet Protocol (IP) addresses
- Assigned networking information
- Computer name and previous name
- Programs scheduled to launch at startup

All this information is retrieved from specific subkey locations and value names as shown in Table III.

TABLE III. SUBKEY AND VALUE NAME FOR GENERAL INFORMATION

Registry Subkey Location	Value Name(s)
HARDWARE\DESCRIPTION\System\BIOS	SystemManufacturer SystemProductName BIOSReleaseDate BIOSVendor BIOSVersion
SYSTEM\CurrentControlSet\Control\Session Manager\Environment	NUMBER_OF_PROCESSORS PROCESSOR_ARCHITECTURE PROCESSOR_IDENTIFIER
SOFTWARE\Microsoft\Windows NT\CurrentVersion	ProductName DisplayVersion
SOFTWARE\RegisteredApplications	{All Value Names in this Subkey} (e.g., File Explorer, Paint, Notepad)
SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall	DisplayName
SOFTWARE\Microsoft\Windows\CurrentVersion\Run	{All Value Names in this Subkey} (e.g., SecurityHealth)
SYSTEM\CurrentControlSet\Control\ComputerName\ComputerName	ComputerName
SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\LogonUI	LastLoggedOnSAMUser LastLoggedOnUser IdleTime
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters\Interfaces\	DhcpIPAddress DhcpSubnetMask DhcpDefaultGateway DhcpServer DhcpNameServer

V. EVALUATION BASED ON REAL-WORLD DATA

In our evaluation, we demonstrate how our custom developed application effectively creates a user's digital footprint by exclusively retrieving registry data from target systems. To achieve realistic results for our assessment, private data is extracted from actual user accounts in several real-world collaborated companies. To corroborate our evaluation, diverse organizations in the fields of accounting, law, and engineering were sampled and utilized for our test cases. Of course, all results shown in this paper are redacted to erase personal or private information from those real-world data.

For our test cases, we executed our application on three Windows 10 pro workstation (we also tested the application on Windows 11 pro and it worked as well). Also, to verify that our custom application evaded other security products, we uploaded and analyzed the executable on VirusTotal.com, a service that uses over fifty malware and breach detection engines (e.g., Avast, AVG, Sophos, Kaspersky, McAfee). The results from the VirusTotal's detection and sandbox analysis reported no threats found.

Initiating our assessment, we run our specialized software, sourced from our GitHub repository "https://github.com/eamoruso/UserProfileAttack", on each of the systems targeted for analysis, in order to collect data and generate the necessary CSV files. In a real-world context, a malicious actor could potentially employ our program on a Windows-based target machine, leveraging avenues like a successful phishing attack [21] where the user is manipulated

into running our specialized application on their Windows computer.

Once the files have been generated, we transfer them to our OneDrive account for further handling. In a malicious context, these files would instead be sent to the perpetrator's computer or cloud storage, where they could be used for unauthorized or nefarious purposes.

In the following we will introduce three test cases, each from different organizations and serving diverse industries. Each test case will show what was collected, processed, and concluded from the captured information. The names of individuals and company names will be redacted, to avoid breach of confidentiality, for each test case. All three test cases had enterprise security software running on their systems. For security and confidentiality, the security product's name is not mentioned.

A. Test Case One

In order to reach the intended system, we established a remote desktop session while utilizing the user's provided login credentials. Within the scope of this experiment, the necessary access information was furnished to us. With the session successfully initiated, we launched the user's web browser and obtained our application from the GitHub repository. Subsequently, the application was executed via a command prompt, enabling us to retrieve our required data from the system's registry. This can be seen in our screen shots shown in Fig. 8 and Fig. 9.

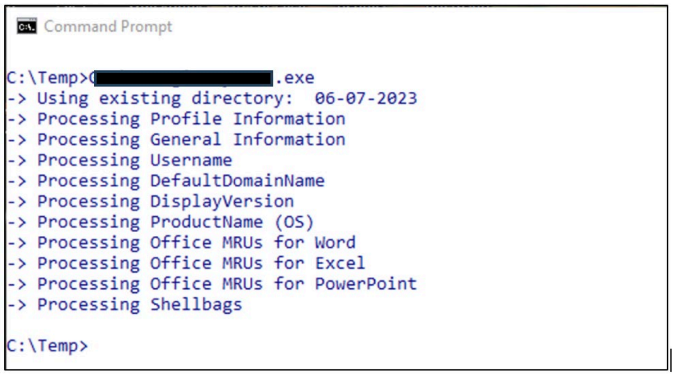


Fig. 8. Running our Custom Developed Application will display processing status as it collects each registry's key data. Private information was redacted.

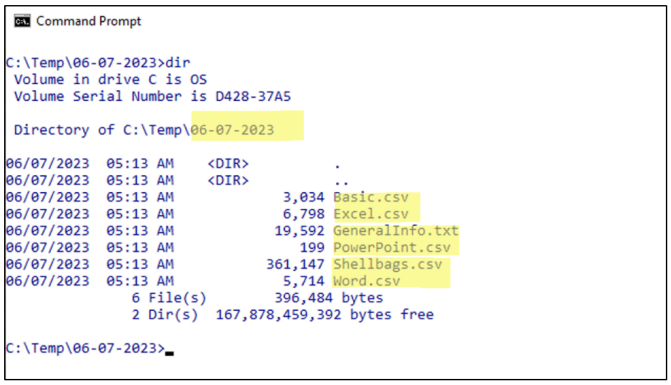


Fig. 9. Example of the directory and files created during program execution.

For this evaluation, we highlight and reveal the information that helps us infer the user's job responsibility, average work hours, and possible position with the organization. To accomplish this, we import the data into Excel, on a local workstation. This can also be accomplished with any other program that supports the CSV file format and provides data manipulation and graphing. With Excel, we can sort and graph our captured information from all the CSV files.

After reviewing both Excel and Word MRUs, we infer the user is either an accountant or bookkeeper since he or she is working mostly with tax return and payroll information, as shown highlighted in Fig. 10 and Fig. 11.

	A
1	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]023 JHL Tax Notes (2).docx
2	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]\2022 Tax Notes 1.docx
3	U:\6 - TANGIBLE CLIENT SERVICES\irs 1099 nec half sheet address.doc
4	U:\1 - WORK IN PROGRESS\Annaul Accounting letter.doc
5	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]S JUNE 2022.doc
6	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\IRS 2[redacted]doc
7	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE[redacted]22 Tax info.doc
8	C:\Users\mshewmaker\Desktop\POA chang[redacted]pdf
9	U:\MARNIE\SSA Forms W-2C amended Short address sheet.doc
10	U:\MARNIE\address Half Sheet.doc
11	S:\3 - ACCOUNTING SERVICES\1 - PAYROLL SERVICES\2022 Payroll Tax Returns[redacted]E
12	S:\1 - TAX SERVICES\2021 Tax Returns\MARNIE\IRS 2021[redacted]l.doc
13	U:\MARNIE\STWMS Letterhead.doc

Fig. 10. The Word.CSV file shows a list of most recently used Word documents. Upon reviewing, it becomes appearant from the filenames "Annaul Accounting letter" and "PAYROLL SERVICES" that the user may have an accounting job.

	A
1	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\Engineered+Machining+LLC_2021.xlsx
2	U:\1 - WORK IN PROGRESS\1 - PAYROLL\STEVE - Angie 2023 PAYROLL LIST.xlsx
3	C:\Program Files\Intuit\QuickBooks Enterprise Solutions 23.0\AlertTemplate.xls
4	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]INVOICE ANALYSIS.xlsx
5	U:\ssheridan\[redacted]II LLC\2022[redacted]VII LLC calculator worksheet.xlsx
6	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\ENGINEERED BANK 2022.csv
7	C:\Program Files\Intuit\QuickBooks Enterprise Solutions 22.0\AlertTemplate.xls
8	\\smscabinet\FileCabinet\1 - TAX SERVICES\2022 Tax Returns\MARNIE[redacted]TATEMENT OF FIN POSITION.xlsx
9	\\smscabinet\FileCabinet\1 - TAX SERVICES\2022 Tax Returns\MARNIE[redacted]TATEMENT OF ACTIVITY 2022.xlsx
10	\\smscabinet\FileCabinet\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]22 BALANCE SHEET & P&L.xlsx
11	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE[redacted]- Installment Sale - Membership Int[redacted]lsx
12	S:\1 - TAX SERVICES\2022 Tax Returns\MARNIE\[redacted]Installment Sale - Membership Int[redacted]xlsx

Fig. 11. Examining the list of entries in the Excel.CSV, we are able to reveal tax and accounting related filenames, for example "2022 Tax Returns."

To help reinforce our claim that the user's job function is most likely an accountant, we look at the General.txt file and

identify tax and accounting software installed on this system, highlighted, and shown in Fig. 12.

	A
27	Title:Uninstalls
28	DisplayName -- CCH Axcess Install and Update Manager
29	DisplayName -- FileCabinet CS Print Driver
30	DisplayName -- Google Chrome
31	DisplayName -- 2020 Information Return System
32	DisplayName -- 2021 Information Return System
33	DisplayName -- 2022 Information Return System
34	DisplayName -- QuickBooks Enterprise Solutions: Accountant Edition 21.0

Fig. 12. The General.TXT file reveals accounting programs "CCH Axcess" and "QuickBooks Enterprise Solutions" available to the target user. The title "Unistalls" means the application is available for uninstalling, meaning it is currently installed.

Using the shellbags information shown in Fig. 13, we can aggregate all the time stamp information and create a frequency graph using Excel. Showing the user's most active time on the system was done by averaging every day's activity and plotting on a twenty-four-hour x-axis. Examining this graph, the user starts work on average between 6am and 8am. It can also be inferred that the user, on average, leaves work between 5pm and 6pm, with occasional after hours, is shown on Fig. 13.

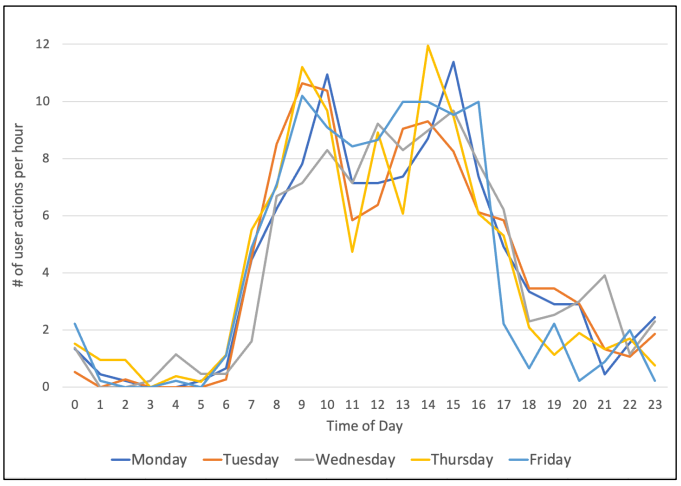


Fig. 13. Daily Computer Usage Activity extracted and saved in the Shellbags.CSV file. [User is an Accountant who still does some work regularly in night time].

B. *Test Case Two*

In this test case, we perform the same process as in the first case. After reviewing Word.CSV, shown in Fig. 14, we were able, with high confidence, infer the user may be a legal professional. Most of the file names contained legal words and are seen in Fig. 14 with keywords such as "Order of Dismissal" and "Memorandum of Law" in the file names.

	A
1	\\Clfserver\cpshare2\CPWin\History\230307_0001\[104084] Letter re Agreed Order of Dismissal re_[redacted].doc
2	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103875] Objection to Notice of Production from Non-Parties.doc
3	S:\[redacted]Lett[redacted]06.02.23_AND.DLD.doc
4	\\Clfserver\cpshare2\CPWin\History\230307_0001\[104058] Letter to Employer - [redacted]terference.docx
5	\\Clfserver\cpshare2\CPWin\History\230307_0001\[104017] Notice of Taking Deposition[redacted]doc
6	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103967] Notice of Taking Deposition[redacted]ende.doc
7	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103324] Memorandum of Law In Opp. to [redacted]ismiss.doc
8	S:\Wholesome[redacted]LC\WT[redacted]NTERFERENCE LETTER [redacted]docx
9	\\Clfserver\cpshare2\CPWin\History\221004_0001\[100698] Notice of Taking Dep[redacted].doc
10	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103676] Order.Writ of Bodily Attachment.doc
11	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103982] Letter re Order Directing Clerk to Issue Writ of B.docx
12	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103962] Motion for Clerk's Default.Eviction.doc
13	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103977] Notice of Filing Returns of Service-Defendants.doc
14	\\Clfserver\cpshare2\CPWin\History\230307_0001\[103538] Summons (5-day).doc

Fig. 14. Word.CSV file shows the list of last modified Word documents by the target user. This information can help identify the user's job responsibilties and important documents on the computer or network.

To help validate if this user was working as a legal professional, we examined the General.TXT file, looking for law related software. The results highlighted in Fig. 15 are programs used by legal professionals to capture billing and case information.

24	Title:Uninstalls
25	DisplayName -- Adobe Creative Cloud
26	DisplayName -- Adobe Genuine Service
27	DisplayName -- Google Chrome
28	DisplayName -- Juris Application
29	DisplayName -- Office 16 Click-to-Run Extensibility Component
30	DisplayName -- Adobe Refresh Manager
31	DisplayName -- Adobe Acrobat
32	DisplayName -- Aderant Total Office Workstation Client

Fig. 15. General.TXT file shows a list of available applciations, two items,"Juris Application" and "Aderant Total Office" gives insight this user has law firm software, helping to further compliment our MRUs and Shellbags findings.

Finally, we reviewed the user's Shellbags.CSV by importing the data into Excel and creating a line graph shown in Fig. 16. From the observed plots, he or she worked a consistent 8am till 5pm schedule. Also, around noon, the user's activity drops, signifying possible lunch break, and then peaks again round 2pm till 5pm.

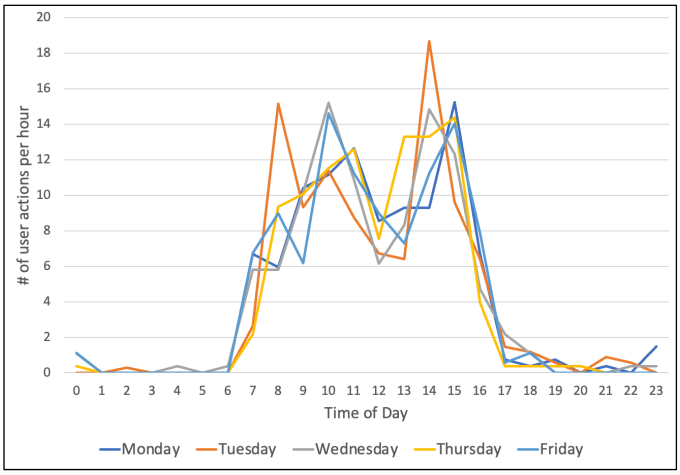


Fig. 16. Daily Computer Usage Activity extracted and saved in the Shellbags.CSV file. [User is a Parralegal Employee].

C. *Test Case Three*

In this last test case, we collect all the information in the same manner as previous cases. Once our files have been acquired, we start evaluating each of the files. Both Word.CSV and Excel.CSV files have very little information, but a few keywords can be found, shown in Fig. 17 and Fig. 18.

	A	B
1	C:\Users\lauren\Desktop\MMS Manual\Madden-Updated.docx	
2	H:\Brandon\Useful Websites\CAD Commands.docx	
3	H:\Brandon\Useful Websites\Useful Websites.docx	
4	H:\MMS project schedule\Projects & Clients\All Souls.docx	
5	C:\Users\lauren\Desktop\Madden-Updated.docx	
6	C:\Users\lauren\Desktop\MMS Manual.docx	
7	C:\Users\lauren\Desktop\Madden.docx	

Fig. 17. In the Word.CSV file we notice the filename "CAD Commands", which means this user is possibly a Computer Aided Design (CAD) operator.

	A
1	H:\Data\21045\Stormwater\21045 BASIN DATA.xlsx
2	C:\Users\lauren\Desktop\PHONE EXTENSION LIST-2023.xlsx
3	H:\Data\21091\Stormwater\21091 Basin Data.xls
4	C:\Users\lauren\Desktop\Job List\JOB-LIST-ACTIVE-NUMERIC.xlsx
5	C:\Users\lauren\Desktop\Job List\JOB-LIST-ACTIVE - BY CLIENT.xlsx
6	H:\MMS project schedule\MMS Upcomming Job schedule chart.xlsx
7	H:\Payton\Design Aids\Calculators.xlsx
8	H:\Payton\Design Aids\Manning Sewer Pipe Calculator.xlsx
9	H:\MMS project schedule\MMS Project Schedule - 2.xlsx
10	H:\Data\22037\Stormwater\Stormwater Calcs.xlsx

Fig. 18. Excel.CSV file lists a few keywords in filenames related to CAD, consistent with what we find in the Word.CSV file in Fig.17.

Next, we start to review the Powerpoint.CSV file and find nothing, meaning the user has never opened or used Microsoft PowerPoint. Now we examine the Basic.CSV, which will contain the user's login name, SID, and installed software. We discover an entry called "Autodesk," a developer of Computer Aided Design (CAD) software, is available to "Lauren" and shown in Fig. 19.

	A	B
46	Waves Audio	29-6-2020
47	lauren S-1-5-21-606991641-2029156010-1706123155-2183	
48	Adobe	20-5-2022
49	AppDataLow	13-5-2022
50	ATI	13-5-2022
51	Autodesk	27-1-2023
52	Bluebeam Software	14-2-2023
53	Canon	20-5-2022
54	Chromium	13-5-2022

Fig. 19. In the Basic.CSV file we are able to identify the user's installed software. Note that "Autodesk" and "Bluebean" are engineering applications useful in CAD design.

To continue the search to identify the user's job function, we examine the GeneralInfo.TXT file. This file also contains the user's "Run at Startup" information, to help identify programs that are executed when the user logs into the system. Again, we see CAD software at startup and hardware typically capable of running engineering applications, shown in Fig. 20.

	A
1	Title:System Environment
2	NUMBER_OF_PROCESSORS -- 8
3	ProductName (OS) -- Windows 10 Enterprise
4	SystemProductName -- Precision Tower 7810
5	Title:Run at Startup (Local Machine)
6	Autodesk Desktop App -- "C:\Program Files (x86)\Autodesk\Autodesk Desktop App\
7	Autodesk Genuine Service -- C:\Program Files\Autodesk\Genuine Service\x64\Ger
8	

Fig. 20. In the GeneralInfo.TXT file, we are able to identify the user's system hardware specifications and startup programs.

Finally, we review the user's Shellbag.CSV file, providing us user's activities. The user is shown to work different hours on different days and can be seen in Fig. 21.

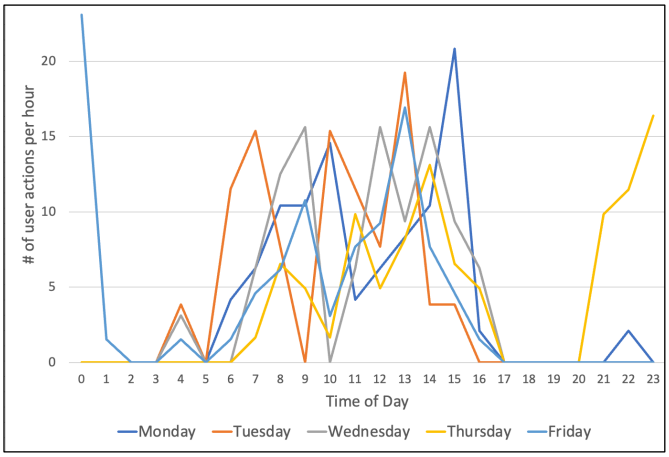


Fig. 21. Daily Computer Usage Activity extracted and saved in the Shellbags.CSV file. [User is a Civil Engineer who works a lot in late night on Thursdays and Fridays].

D. *Summary*

In summary, the three real-world test cases provide a good inference on user's job functions. Having multiple indicators, such as MRUs of documents, spreadsheets, presentations, and installed applications helped establish the user's digital footprint, such as their software in use and essential documents.

More interestingly, building the user's working schedule provides insight into the culture of each profession. For example, the paralegal who worked consistently from 8am to 5pm, Monday through Friday; or the civil engineer that worked irregularly during the daytime and worked extended hours late at night in a couple of weekdays.

During our evaluation of the MRUs, we were alarmed to find a certain special files. Out of the three test cases, two users had documents named "passwords" on their network drive. Having worked in the IT industry for over 10 years, it was a surprise this practice is still used in today's security landscape. This reinforces the importance of this research to expose the dangers of effortlessly capturing a user's profile by running our custom application.

VI. Limitations and Future Work

A. Administrative Privilege

In the context of the Windows operating system, a user account lacking administrative privileges is restricted from accessing registry data belonging to other users on the same local system. At present, our methodology is constrained to demonstrating feasibility. We achieve this by launching our custom application from the account of a designated target user, enabling us to retrieve the digital traces associated with that user.

An alternative approach involves either employing process injection into a running program with administrative privileges or compromising the system to attain privileged access [22]. Through such means, a malicious program crafted by an attacker could potentially provide us with entry to the registry data of all user accounts on the system, enabling a more impactful attack to retrieve digital footprints of all existing users.

B. Future Work

Although the Windows registry holds an extensive volume of information, it does not retain specific details such as browsing history or exact timestamps of user interactions on websites. Nevertheless, a substantial proportion of users dedicate a significant amount of their leisure and working hours to browsing the Internet, accessing various websites. As a result, the way browsers are used and the sites visited assume a crucial role in shaping a user's digital footprint. In our future work, we plan to incorporate a feature that enables custom or third-party applications to extract data linked to an Internet browser. Currently, the alternative approach involves utilizing a third-party utility to access these files. One such application is available from NirSoft's web page and is accessible as freeware [23].

One more advantageous factor that could contribute to an individual's digital footprint is the incorporation of the system's user login and logout events. Although a portion of this data is stored in the registry, it qualifies as sensitive information and thus requires heightened privileges from the operating system. Presently, the recommended approach is to make use of the "Event Viewer," a native Windows tool. In our forthcoming research, we will delve into the application of the Windows Event Log API and its integration within our customized software application.

VII. Conclusion

In this paper, demonstrated by our custom developed application, we revealed that it is possible to expose the digital footprint of a user's account running on a Windows machine by solely accessing the Windows registry. Alarmingly, this was achieved without triggering any security mechanisms. Through the use of native Windows APIs in our programming, we identified a security flaw that enabled us to access a range of information from the system registry. This data covered aspects like recently opened files, timestamps related to system activities, network configurations, hardware specs, and software preferences. Such information could be leveraged to stage a sophisticated cyber-attack. Additionally, a malicious actor could glean both cyber and non-cyber personal information about the user, including work hours, lunch intervals, days off, and job functions.

Acknowledgment

This research was sponsored by the U.S. National Science Foundation (NSF) under Grant DGE-2325452.

References

[1] E. Bott, *Windows 10 IT Pro Essentials Top 10 Tools*. First Edition (1st. ed.). USA,: Microsoft Press, 2016. ch. 9, pp. 128-129.

[2] R. Kazanciyan, M. Hastings. "Investigating powershell attacks." Mandiant. Accessed: Feb. 7, 2023. [Online]. Avaliable: https://www.blackhat.com/docs/us-14/materials/us-14-Kazanciyan-Investigating-Powershell-Attacks-WP.pdf

[3] W. Munroe, "Why Malicious Actors Love PowerShell Attacks and How to Defend Them." Rangeforce.com. https://www.rangeforce.com/blog/powershell-attacks-and-how-to-defend-them (accessed Feb. 13, 2023).

[4] T. Wojewoda. "Hunting and Gathering with PowerShell," SANS Institute., White Paper, 2019. [Online]. Available: https://www.giac.org/research-papers/38842/ (accessed Feb. 14, 2023).

[5] Y. Zhu, P. Gladyshev, and J. I. James, "Using shellbag information to reconstruct user activities," *Digital Investigation*, vol. 6, pp. S69–S77, Sep. 2009, doi: 10.1016/j.diin.2009.06.009.

[6] R. Mize, "Behavior of Shellbags in windows 10." Ph.D. dissertation, Utica College, 2018. [Online]. Available: https://www.proquest.com/dissertations-theses/behavior-shellbags-windows-10/docview/2108990957/se-2

[7] A. Đuranec, D. Topolčić, K. Hausknecht and D. Delija, "Investigating file use and knowledge with Windows 10 artifacts," 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 2019, pp. 1213-1218, doi: 10.23919/MIPRO.2019.8756877.

[8] "Publicly Available Tools Seen in Cyber Incidents Worldwide," *Cybersecurity and Infrastructure Security Agency CISA*, Accessed: Feb 14, 2023. [Online] Available:

https://www.cisa.gov/news-events/cybersecurity-advisories/aa18-284a

[9] R. Cobb. "Entering a Covenant: .NET Command and Control." Specterops.io., Accessed: Apr. 22, 2023 [Online]. Available: https://posts.specterops.io/entering-a-covenant-net-command-and-control-e11038bcf462?gi=ce1cb24bd25a

[10] A. Barakat and A. Hadi, "Windows Forensic Investigations Using PowerForensics Tool," *2016 Cybersecurity and Cyberforensics Conference (CCC)*, Amman, Jordan, 2016, pp. 41-47, doi: 10.1109/CCC.2016.18.

[11] S. Zavala, N. Shashidhar and C. Varol, "Cybersecurity Evaluation with PowerShell," in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, Beirut, Lebanon, 2020, pp. 1-6, doi: 10.1109/ISDFS49300.2020.9116258.

[12] A. Singh, H. S. Venter, and R. A. Ikuesan, "Windows registry harnesser for incident response and digital forensic analysis," *Australian Journal of Forensic Sciences*, vol. 52, no. 3, pp. 337–353, Dec. 2018, doi: 10.1080/00450618.2018.1551421.

[13] Y. Zhu, P. Gladyshev, and J. I. James, "Using shellbag information to reconstruct user activities," *Digital Investigation*, vol. 6, pp. S69–S77, Sep. 2009, doi: 10.1016/j.diin.2009.06.009.

[14] A. D. Kent and L. M. Liebrock, "Differentiating User Authentication Graphs," 2013 IEEE Security and Privacy Workshops, San Francisco, CA, USA, 2013, pp. 72-75, doi: 10.1109/SPW.2013.38.

[15] "Windows Registry Cheat Sheet," www.13cubed.com. https://www.13cubed.com/downloads/windows_registry_cheat_sheet.pdf (accessed Feb 11, 2023).

[16] F. Korkmaz, "Windows Artifacts - Fahri Korkmaz - Medium," *Medium*, Jan. 07, 2022. [Online]. Available: https://r4bb1t.medium.com/windows-artifacts-8fae778aa8c7

[17] V. D. Munister, A. L. Zolkin, A. V. Ishkov, O. V. Kosnikova, and I. A. Poskryakov, "Computational analysis of the digital footprint using machine learning and artificial intelligence," *Journal of Physics*, vol. 2094, no. 3, p. 032003, Nov. 2021, doi: 10.1088/1742-6596/2094/3/032003.

[18] R. Yasin, "Stealing Data By 'Living Off The Land,'" *Dark Reading*, Sep. 03, 2015. [Online]. Available: https://www.darkreading.com/analytics/stealing-data-by-living-off-the-land-

[19] "Companies using Microsoft Office 365." Enlyft.com. https://enlyft.com/tech/products/microsoft-office-365 (accessed May 30, 2023).

[20] Helenclu, et al. "How to reset user options and registry settings in Word - Microsoft 365 Apps," *Microsoft Learn*, May 05, 2022. https://learn.microsoft.com/en-us/office/troubleshoot/word/reset-options-and-settings-in-word (accessed May 30, 2023).

[21] R. Alabdan, "Phishing Attacks Survey: Types, vectors, and technical Approaches," *Future Internet*, vol. 12, no. 10, p. 168, Sep. 2020, doi: 10.3390/fi12100168.

[22] A. Mohanta and A. Saldanha, "Code injection, process hollowing, and API hooking," in *Apress eBooks*, 2020, pp. 267–329. doi: 10.1007/978-1-4842-6193-4_10.

[23] BrowsingHistoryView. (2023), NirSoft, [Online]. Available: https://www.nirsoft.net/utils/browsing_history_view.html