



Digital forensic analysis of discord on google chrome

Khushi Gupta*, Cihan Varol, Bing Zhou

Sam Houston State University, Huntsville, TX, USA

ARTICLE INFO

Article history:

Received 3 September 2022

Accepted 31 October 2022

Available online 28 November 2022

Keywords:

Digital forensics

Discord

Social media

Web browser forensics

ABSTRACT

Over the past decade, social media applications have significantly increased their market share and garnered a wide user base. However, these applications have also attracted the attention of criminals desiring to exploit the apps to support illicit operations due to the low barrier to entry and ease of usage. A digital forensic investigation of these applications can reveal valuable information about criminal activity and the suspect. Discord is a Voice over Internet Protocol (VoIP) service that enables text, image, video, and audio chats. It has grown in popularity, and as a result, it is subject to increased use by cybercriminals. In this paper, we examine the remnants of the increasingly popular social media application “Discord” when used on the Google Chrome web browser. We recovered various artifacts such as payment information, sent messages, account settings, conversations, uploaded attachments, and much more, all of which could be utilized in a forensic investigation.

© 2022 Elsevier Ltd. All rights reserved.

1. Introduction

The widespread usage of the Internet has rapidly transformed the means of communication people use. Social Media (Voice over IP (VoIP) and instant messaging) applications have now become the most prevalent means of communication. These applications provide channels for community-based collaboration, interaction, and content-sharing. Social media is a collection of websites and applications built to enhance networking and sharing information online Power (2014). Evidence obtained from social media applications is the new frontier for digital forensics. The plethora of social media applications combined with hundreds of thousands of its users generate large amounts of transmissions daily, yielding a lot of digital evidence for forensic analysis. 4.48 billion people actively used social media applications in 2021, which was a 13.1% increase from 3.69 billion in 2020, indicating the projected increase in these numbers over the years Dean (2021).

Numerous studies have been conducted on some of the most widely used social media platforms, including Facebook, WhatsApp, TikTok, Skype, Twitter, and others Pandela and Riadi (2020), Chang and Yen (2020), Paligu and Varol (2020), Shao et al. (2019). Despite the popularity of these applications, many new social media applications are being developed and are also attracting

millions of users, such as Discord. Discord is a social media platform that allows users to communicate with voice and video calls, text messages, media, and files through private chats or as a part of a community called “servers,” all by remaining completely anonymous. Discord can be accessed via a web browser or by installing the application. In 2021, Discord had about 390 million registered users with 150 million active monthly users Campbell (2022). Due to its anonymity feature, it can attract many criminals as part of its growing community, and thus, it can be used for nefarious purposes.

According to Discord’s Transparency report of June 2021, community guideline breaches amounted to more than 430,000 Safety (2021). This report addresses violations within the Discord application community, and most of the reports concern actions such as harassment, cybercrime, graphic content, spam, and exploitative content. Additionally, Discord has been previously used to commit many crimes, some of which include selling illicit products such as stolen credentials for Delta airlines and Hilton hotels Brewster (2019), Patterson (2019) and malware used to infect computer networks and to distribute child pornography UnitedStatesAttorney’sOffice (2021). These activities have also resulted in Discord being investigated by the FBI Brewster (2019). According to The National White Collar Crime Centre (NW3C), Discord demonstrates great forensic value to law enforcement due to its explosive growth, which shows no signs of slowing down NW3C (2017).

Users perform several activities such as surfing the Internet, downloading files, using social media applications, and accessing

* Corresponding author.

E-mail addresses: kxg095@shsu.edu (K. Gupta), cxv007@shsu.edu (C. Varol), bxz003@shsu.edu (B. Zhou).

their e-mail accounts on a web browser [Sonmez and Varol \(2019\)](#). Web browser forensics attempts to look for residual data/artifacts left on a web browser through web activities. This is due to the fact that many criminal lawsuits are based on the evidence obtained from a user's web activity [Sonmez and Varol \(2019\)](#). Despite Discord being used for various crimes and its growing number of users, little research has been done investigating the platform with respect to a web browser. This research, therefore, aims at uncovering the digital artifacts recovered from the Discord application on the Google Chrome web browser on Windows 10. We target the Google Chrome web browser as it is the most used browser among other web browsers such as Internet Explorer, Mozilla Firefox, Safari, and Opera [Oberlo \(2022\)](#).

We believe that Discord can be a vital data source for digital evidence in digital forensic investigations. The digital evidence that can be obtained includes but is not limited to text posts, participants in a server, images, geo-location data, and so forth. The results presented can then be used as a reference point for other researchers on social media forensics and by digital forensic analysts to understand what digital evidence can be discovered from Discord on web browsers.

The rest of the paper is organized as follows. Section 2 provides background information on Discord forensics. In section 3, we detail the methodology followed in this study, while in section 4, we present our findings. In section 5, we discuss the relevance of our results, and lastly, we conclude our work and mention possible directions for future study in section 6.

2. Related work

Unfortunately, the numerous benefits of social networking come with a number of risks. Cybercriminals now have additional opportunities to carry out their unlawful activities due to the rise of online social networking platforms [Patel et al. \(2017\)](#). Even though Discord was created for gaming, numerous communities have adopted it to communicate information. While much research has been conducted around the most used social media applications such as WhatsApp, Instagram, Facebook, Twitter, etc., more research needs to be focused on the Discord application. Discord has become a good source of digital evidence due to its widespread adoption. In this section, we will discuss previous work conducted on Discord applications as well as some other related work.

2.1. Social media applications other than discord

Most research on social media applications has been generalized to provide fundamental knowledge about social media applications in digital forensic situations. These articles lay the foundation from which we can build upon.

Social media proves to be an abundant source of evidence and information about potential suspects, victims, and witnesses. It acts as a dynamic source of information created by individuals, such as images, videos, geolocation data, friends, text posts, and so forth. The work of [Arshad et al. \(2019\)](#) corroborates that evidence from social media platforms has a lot of potential to give exceptional support to investigators during the criminal investigation process. Adding to that, [Baggili and Breitingner \(2015\)](#) looked into the problems with data sources used in cyber forensics. They talked about how social media could affect future forensic investigations and advance data availability in cyber forensics. Investigators can access artifacts such as text messages, friends and friend groups, photos, and geolocation data using social media forensics.

Other research works that are more focused on specific applications are as follows. [Pandela and Riadi \(2020\)](#) conducted browser forensics on the TikTok web browser application using a case

scenario of defamation videos being uploaded on the social media application using a chrome browser and later deleting it. The results revealed that the text post caption, suspect's username, profile photographs, and video post were successfully recovered.

Touching on the same aspect, [Chang and Yen \(2020\)](#) conducted digital forensics on LinkedIn using three different web browsers (Google Chrome, Mozilla Firefox, and Microsoft Edge) running under the Windows 10 operating system. The study found a lot of valuable evidence such as account and password information, friend lists and requests, evidence of a post, the post being liked, and the comment on the post that was recovered from the browsers. A comparison was also made, concluding that all the web browsers yielded the same types of evidence.

Lastly, [Paligu and Varol \(2020\)](#) use a single case pretest-posttest quasi-experiment to analyze the artifacts left from the widespread application Whatsapp, when used on the Google Chrome web browser. The authors investigate the IndexedDB storage utilized by Google Chrome to recover artifacts from the Whatsapp web application (Whatsapp Web). The research revealed that IndexedDB stores significant artifacts from the Whatsapp web application. This residual data has great forensic value as it can be used for time frame analysis during forensic investigations. Additionally, a proof of concept tool called BrowSwEx was developed to demonstrate the value of the artifacts found and for time frame analysis.

2.2. Discord

Some of the past research on the Discord application includes an in-depth analysis of the Discord cache structure to develop a Python tool, called DiscFor, to automatically recover any relevant forensic artifacts from the Discord client application on Windows 10, macOS, and Linux systems [Motylinski et al. \(2020\)](#). The authors focused on analyzing the Discord application's caching structure and noted that Discord on Windows 10 and macOS uses a Chromium Disk Cache, while on Linux, Discord uses a Simple Cache structure. The authors then used these cache structures and activity log files from Discord to build the DiscFor tool, facilitating the automatic extraction of all information stored on these Discord local files.

Another recent research involving Discord was done by [Moffitt et al. \(2021\)](#). They conducted a forensic investigation of the Discord application in the Windows 10 operating system and attempted to discern what artifacts were available from the forensic study in terms of forensic merit and cybersecurity. The results report the findings of logs of plaintext messages, the email address, the fingerprint and authentication token, and unencrypted images, videos and GIFs kept in the local storage. The paper then discusses the security aspects of locally stored information and provides guidelines for forensic investigators and users concerning data privacy.

Furthermore, [Davis, McInnes and Ahmed \(2022\)](#) present a memory forensic examination of instant messaging applications on the Linux OS, namely, Discord and Slack. They took memory snapshots after conducting specific activities on the two platforms, which revealed a wealth of information. Some artifacts recovered were usernames, emails, passwords, messages, conversations, and files from the system and app memory dumps which would be very helpful for forensic investigators.

The literature review discussed in this section has helped us identify the gaps in the current literature regarding forensic analyses of the Discord application. To the best of our knowledge, only a few studies have tackled the Discord application with regard to digital forensic analysis. Furthermore, research has yet to tackle complete browser forensics for the Discord application [Moffitt et al. \(2021\)](#). Therefore, our study targets browser forensics on the Discord application with the specific aim of recovering the residual artifacts left behind through its usage on the Google Chrome web browser.

3. Methodology

This research is based on finding artifacts left by Discord on the Google Chrome browser using a single case pretest-posttest quasi-experiment formed from the principles described by [Cook and Campbell \(1979\)](#). A quasi-experiment is a research design that attempts to establish a cause-and-effect relationship. It contains just one subject that is measured before the treatment, put through the treatment, and then measured again for change evaluations. Based on this, a set of measurements are applied before and after using the Discord application on the browser. The artifacts that were exclusively filled by the treatment were determined based on the variations noticed from the results of the measurements.

- **Pretest:** Data stored by the Google Chrome browser upon being downloaded is extracted and evaluated to obtain the artifacts inherently present in the storage location of the application.
- **Treatment:** A set of Discord use cases created from common user behaviors was carried out to populate the artifacts stored by Google Chrome.
- **Posttest:** After the treatment, data is extracted from the same location as in the pretest stage. This is done to compare the artifacts from the pretest stage and isolate the artifacts that resulted from the treatment.

To provide a controlled environment for Discord, we created three new accounts for this study: Jdoe67160 (PC1), kimkarter (PC2), and alphaben (PC3). They were then personalized through different profile pictures. These accounts were solely utilized on their computers, and each computer was separated, with no direct connections between them other than through the Discord application. All the computer systems were formatted and installed with Google Chrome to ensure no residual data from the previous usage of chrome affected the experiment. To sign up for Discord and other social media accounts, Google accounts were also created for the three users. The Google accounts enabled us to create accounts with Facebook, Twitch, GitHub, and YouTube, which can then be linked to the Discord profiles.

The experimental setup thus included three systems, one for each user. A Dell Precision 5820 Tower (PC1) that operates a windows 10 professional 64-bit single language operating system with Google chrome version 99.0.4844.51 (official build) (64-bit) browser installed. The central processing unit is Intel(R) Xeon(R) W-2123 @ 3.60 GHz with a memory size of 80 Gigabytes.

The second system (PC2) is a Dell Optiplex 9020 that operates a windows 10 professional 64-bit single language operating system with Google chrome version 99.0.4844.51 (official build) (64-bit) browser installed. The central processing unit is Intel(R) Core(TM) i7-4790 @ 3.60 GHz with a memory size of 8 Gigabytes.

The third system (PC3) is a virtual machine that is hosted on a Dell Inspiron 15 laptop that operates Windows 10 professional 64-bit single language operating system with Google chrome version 99.0.4844.51 (official build) (64-bit) browser installed. The central processing unit is Intel(R) Core(TM) i7-8550U @ 1.80 GHz with a memory size of 8 Gigabytes. This machine is the host to a virtual machine created using a VMware workstation. The virtual machine, running a Windows 10 Professional operating system, is installed with 4 Gigabytes of RAM.

3.1. Research questions

The following are some of the research questions addressed by the study.

1. Can we find any artifacts from the Discord application when it is used on the Google Chrome browser, if so, does it carry significant value for a digital forensic investigation?
2. Is there any difference between the artifacts found when chrome is used on a virtual machine?
3. What personally identifiable information of the users can be recovered?
4. What information is stored in the browser cache?
5. What are the locations in which residual data is stored?

By answering these research questions, this research investigates the different residual data/artifacts and how they can be used as evidence during forensic investigations.

3.2. Pretest

The default/initial artifacts present in the Google Chrome storage of all the computer systems were tested before the treatment was carried out. The procedure carried out was as follows:

- The Google Chrome web browser was initiated without logging into a google account
- No browsing was done and the browser was left idle for 10 min
- The artifacts were then extracted and analyzed from the persistent storage locations used by Google Chrome

3.3. Treatment

The experimental methodology of this research are the activities performed with Discord on the web browser to create artifacts. A set of use cases were designed as the treatment to populate the artifacts. The activities performed in this experiment are in accordance with common user behavior with web browsers and messenger communication applications. The following activities are the most common activities on the Discord application.

- One on One chat with friends
- Formation of Groups
- Creation of a public/private Discord server
- Using emojis and Hyperlinks while chatting
- Sending digital media (Pictures, Videos, GIFs) while chatting
- Audio and Video call with friends individually or as a group
- Muting and turning off camera during an audio and video call respectively
- Edit, delete, reply, and pin messages
- Copy message link for sharing purposes
- Linking the Discord account to various services such as YouTube, Twitch, GitHub, Facebook, Spotify, etc.
- Adding a payment method
- Downloading Files
- Using slash commands to make a spoiler message

3.4. Data acquisition

A Windows computer stores browser files created by Google Chrome in " OS Drive:\Users*(username)*\AppData \local\Google \Chrome\UserData \Default". Google Chrome saves browser data in different mechanisms like Bookmarks, Cache, Cookies, History, Preferences, Visited Links, Top Sites, etc. The storage locations that will be analyzed to recover Discord artifacts in this experiment include history, cache data, downloaded files, form values, and local storage. Most Google Chrome files are saved in SQLite database format, while some files also use LDB (.json) format.

3.4.1. History

One of the most typical targets for a forensic investigation is browser history. It provides information about all the websites that have been visited, including metadata such as the date, time, and website address. As a result, it is a crucial place to look into for investigators from a digital forensic standpoint.

3.4.2. Cache data

Cache is a web browser storage mechanism that stores data frequently transferred to the client to reduce the computing resources utilized for the transfer. Browsers save such information from the websites so it can be used without requesting it again. Today's forensic technologies employ data from the cache to recreate web pages for analysis. Reconstruction of websites is forensically beneficial since the websites may be seen in the same way that the suspect saw them.

3.4.3. Form values

Google Chrome stores form/web data in an SQLite format in the web data file. This file may contain sensitive user information such as usernames and passwords, addresses, and payment information, among others, which might be saved by the user for easy data entry. This can provide crucial information about the suspect during an investigation.

3.4.4. Downloaded files

Google Chrome saves the information about downloaded files in an SQLite-formatted History file. The download information, as well as browser activity information, is stored in the history file. Each URL has a visit count linked with it, which indicates how many times it has been visited.

3.4.5. Web storage

Web storage consists of local and session storage. For this experiment, our primary focus will be on retrieving artifacts from the local storage since local storage is not as volatile as the session storage. Local storage is a storage location managed by the web browser, and as the name suggests, it stores all the information locally on the machine in which the browser application is installed. The local storage can be accessed through the Google Chrome developer tools (G) or in the form of .log files, which can be found in the folder where Google Chrome stores its files. JavaScript object notation (JSON) is a data format for storing and retrieving data quickly, and humans can easily read and interpret it without additional processing.

Table 1 below summarizes the locations and the file formats of the browser web storage for the Google Chrome browser that were used in this paper.

Two tools were used to analyze the Google Chrome files to retrieve digital artifacts linked to the usage of the Discord application on the web browser. We used DB Browser (SQLite) to view the SQLite Google Chrome files, whereas ChromeCacheView was used to parse the Google Chrome cache files.

3.5. Use case scenarios

Based on the most common behavior of users on Discord, three use case scenarios were devised for the experiment. Several types of interactions were conducted among the three users in the scenarios. Additionally, all the activities performed in each use case scenario were timed for better analysis of evidence. The following are the use case scenarios used for the experiment. Detailed use case scenarios 1 and 2 can be found in Section 7, Appendix.

- Scenario 1: Text-only private conversation amongst PC1 and PC2
- Scenario 2: A group conversation with digital media (pictures, emojis, GIFs, and videos) among all users.
- Scenario 3: All accounts having a conversation within a controlled server.

Preliminary settings: All three accounts were first set up in the respective systems. The two other users were then added as friends for each account. For all the accounts, the default Discord user account settings were used. However, the account configured on PC1 (Jdoe67160) has some preliminary settings configured. These settings include linking a Facebook, Twitch, and YouTube account to the Discord account, adding a payment method on this account for in-app purchases, and turning on desktop notifications.

4. Results

The treatment for this experiment was using the Discord application on the Google Chrome web browser. The results are based on the different treatment activities performed throughout the experiment. This resulted in the creation of many artifacts obtained from different storage mechanisms, as highlighted in this section. The results obtained from the experiment are evaluated based on the artifact value for forensics investigations. The artifacts we deemed significant are listed in this section with the corresponding treatment activity and the storage mechanism it was retrieved.

Some of the artifacts were extracted from local storage, which keeps an implementation of LevelDB. Storage files with the extensions .ldb and .log can be found in this location. The rest of the files, such as the manifest file, are support files that let the .ldb and .log extension files maintain track of versions and exchange data. The log files contain UTF (Unicode Transformation Format-16) encoded information and binary entries. Adding on, during the experiments, it was observed that the .log files were not deleted after the termination of the browser. Thus, we conveniently obtained the artifacts from the clear text-based .log files using a Hex editor.

Google Chrome cache also provided us with various forensic artifacts. Cache gave us access to files generated by Discord, as well as files directly uploaded by Discord users. Generated files include JavaScript files, chat logs, and other application content such as images, stickers, and emojis. Any file stored in the cache contains several headers that provide valuable information on the file. Some of the most important headers from the perspective of a forensic investigation are listed as follows.

- Content type: Multipurpose Internet Mail Extension (MIME) content type is a standard that indicates the format of a document. It consists of type (such as text) and subtype (such as text/plain, image/png, and video/mpeg).
- ETag: This is a unique identifier of the specific version of the resource.
- Response time: This indicates the time when the requested resource was loaded the last time.
- Last modified time: This indicates the time when the resource was last modified.
- Server name: This indicates the name of the server hosting the service.
- Server IP: This lists the IP address of the server hosting the service.

4.1. Account creation and settings

A lot of information was garnered through the account creation

process. This information can reveal a great deal of personal information about the account holder and thus, is extremely valuable during a forensic investigation. Table 2 shows the different artifacts recovered in regard to the account creation and its settings from local storage, cache, and Google developer tools.

Fig. 1 shows a local storage file from PC2. Some of the forensic artifacts recovered are the unique user ID associated with kimkarter, his username, a four-digit random unique discriminator, also known as a Discord tag which is used to identify a user, and multi-account settings that can help deduce if multiple accounts were logged in Discord at any given time. Additionally, Fig. 2 depicts another snippet of the same file shown in Fig. 1 with artifacts such as locale, status, theme, time zone offset, custom status, expiration date and time, and the unique Discord token associated with the login of the account. The Discord login token is a crucial artifact as it acts as a user's login details. Thus, a Discord token can be used to log in to a user's Discord account without needing login details. Another artifact found in local storage is "Streamer mode" settings, as shown in Fig. 3. Discord launched its streamer mode to prevent the leaks of personal and sensitive data of a user while publicly streaming on the platform.

Figs. 4–6 shows the artifacts recovered from PC2 cache files. Fig. 4 shows a cache file named "true.json" that presents some of the account-related artifacts mentioned in Table 2. Fig. 5 shows the "true.json" file of kimkarter's Discord friend alphaben (PC3). This figure is very similar to Fig. 4 in terms of the artifacts presented, with the only difference being that it lists "mutual_guilds" which is a unique id of a server that both kimkarter and alphaben are participants of. Lastly, Fig. 6 presents the "relationship.json" cache file which lists information about all the Discord friends that kimkarter (PC2) has.

Finally, Fig. 7 depicts some of the artifacts recovered from the Google developer tools. Most of the artifacts recovered are similar to those retrieved from the cache files and local storage files, but the information revealed through Google developer tools is volatile, which means that this information will no longer be accessible after the Chrome tab is closed. Chrome developer tools can give a vast amount of information if the computer is switched on during analysis.

Table 1
Folder location and file format of the web storage technologies.

Browser Web Storage	Folder Location	File Format
History (H)	C:\Users\<username>\AppData\Local\Google\Chrome\User Data\Default\History	.sqlite
Cache Data (C)	C:\Users\<username>\AppData\Local\Google\Chrome\User Data\Default\Cache	.sqlite
Form Values (F)	C:\Users\<username>\AppData\Local\Google\Chrome\User Data\Default\Web Data	.sqlite
Downloaded Files (D)	C:\Users\<username>\AppData\Local\Google\Chrome\User Data\Default\Local Storage	.json

Table 2
Account creation and settings and its associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Account creation and settings	User ID	L, C
	Username	L, C
	Discriminator	L, C
	Multi-account settings	L
	Locale	L, C, G
	Status and Custom Status	L, G
	Status Expiry	L, G
	Theme	L, G
	Time zone offset	L, G
	Token	L, C
	Email	C
	Verification Status	C
	Phone	C
	Relationships	C
	Email settings	C
	Streamer mode settings	L

User account settings in the cache can be found in the files with the URL that starts from "<https://Discord.com/api/v9/users/@me>". For example, user email settings will be found in the file with the URL "<https://Discord.com/api/v9/users/@me/email-settings>".

4.2. Account login

Account logins can provide a forensic investigator with many valuable artifacts when it comes to creating a timeline in an investigation and laying out the series of events after the log-on. Table 3 lists some of the artifacts found with regard to account logins in Discord. Fig. 8 shows a snippet of the login token encryption time in Unix time and hash values as recovered from the local storage file of PC2 depicting when the user logged in. To convert Unix time format to a human understandable form, we used online Unix time converters.

4.3. Payment method

One of the preliminary settings on PC1 before conducting the use case scenarios was to add a payment method to Discord. Upon analyzing the Google Chrome cache (payment_sources.json), several artifacts were found with respect to adding a payment method, as highlighted in Table 4. Artifacts that originated as a result of adding a payment method to Discord are shown in Fig. 9. Highly sensitive and confidential information such as the last four digits, expiry month and year, and billing address are exposed with no form of encryption which can definitely be a game changer in an investigation. Fig. 9 contains highly sensitive information; thus, it has been blurred.

4.4. Connections

Discord has a feature that lets one connect their Discord account to other accounts. Connections refer to the different third-party accounts that have been linked to Discord, such as YouTube, Twitch, and Facebook. Table 5 lists the different account connection artifacts uncovered from cache file, "connections.json".

Fig. 10 shows the different account connections the user Jdoe67160 (PC1) has alongside other different artifacts such as the type of connection, User ID, name, activity status settings, verification status, friend sync status, and access token. The account connections of PC1 based on the preliminary settings can be confirmed by looking at Fig. 11. The list of connections to third-party accounts can also be recovered from the file "true.json," which uncovers the artifacts related to account settings as depicted

```

00004180 01 01 7B 22 5F 73 74 61 74 65 22 3A 7B 22 75 73 {"_state":{"us
00004190 65 72 73 22 3A 5B 7B 22 69 64 22 3A 22 39 36 33 ers":[{"id":"963
000041A0 38 38 36 33 38 36 35 31 38 33 30 32 37 32 30 22 886386518302720"
000041B0 2C 22 61 76 61 74 61 72 22 3A 6E 75 6C 6C 2C 22 ,"avatar":null,"
000041C0 75 73 65 72 6E 61 6D 65 22 3A 22 6B 69 6D 6B 61 username":"kimka
000041D0 72 74 65 72 22 2C 22 64 69 73 63 72 69 6D 69 6E rter","discrimin
000041E0 61 74 6F 72 22 3A 22 39 31 31 31 22 2C 22 74 6F ator":"9111","to
000041F0 6B 65 6E 53 74 61 74 75 73 22 3A 32 7D 5D 2C 22 kenStatus":2}],
00004200 69 73 4D 75 6C 74 69 41 63 63 6F 75 6E 74 50 72 isMultiAccountPr
00004210 65 76 69 6F 75 73 6C 79 45 6E 61 62 6C 65 64 22 eviouslyEnabled"
00004220 3A 74 72 75 65 7D 2C 22 5F 76 65 72 73 69 6F 6E :true}, "_version
00004230 22 3A 30 7D 01 2A 5F 68 74 74 70 73 3A 2F 2F 64 ":0} *_https://d

```

Fig. 1. Account settings artifacts recovered from local storage.

```

00004590 22 3A 22 65 6E 2D 55 53 22 2C 22 73 74 61 74 75 ": "en-US", "statu
000045A0 73 22 3A 22 6F 6E 6C 69 6E 65 22 2C 22 74 68 65 s": "online", "the
000045B0 6D 65 22 3A 22 64 61 72 6B 22 2C 22 72 65 73 74 me": "dark", "rest
000045C0 72 69 63 74 65 64 47 75 69 6C 64 73 22 3A 5B 5D rictedGuilds": []
000045D0 2C 22 64 65 66 61 75 6C 74 47 75 69 6C 64 73 52 , "defaultGuildsR
000045E0 65 73 74 72 69 63 74 65 64 22 3A 66 61 6C 73 65 istricted": false
000045F0 2C 22 66 72 69 65 6E 64 53 6F 75 72 63 65 46 6C , "friendSourceFl
00004600 61 67 73 22 3A 7B 22 61 6C 6C 22 3A 74 72 75 65 ags": {"all": true
00004610 7D 2C 22 66 72 69 65 6E 64 44 69 73 63 6F 76 65 }, "friendDiscove
00004620 72 79 46 6C 61 67 73 22 3A 30 2C 22 67 75 69 6C ryFlags": 0, "guil
00004630 64 50 6F 73 69 74 69 6F 6E 73 22 3A 5B 5D 2C 22 dPositions": [], "
00004640 67 75 69 6C 64 46 6F 6C 64 65 72 73 22 3A 5B 5D guildFolders": []
00004650 2C 22 74 69 6D 65 7A 6F 6E 65 4F 66 66 73 65 74 , "timezoneOffset
00004660 22 3A 33 30 30 2C 22 63 75 73 74 6F 6D 53 74 61 ": 300, "customSta
00004670 74 75 73 22 3A 7B 22 74 65 78 74 22 3A 22 49 74 tus": {"text": "It
00004680 73 20 4B 69 6D 21 22 2C 22 65 78 70 69 72 65 73 s Kim!", "expires
00004690 41 74 22 3A 22 32 30 32 32 2D 30 34 2D 32 34 54 At": "2022-04-24T
000046A0 30 35 3A 30 30 3A 30 30 2B 30 30 3A 30 30 22 7D 05:00:00+00:00"}
000046B0 2C 22 63 6F 6E 74 61 63 74 53 79 6E 63 55 70 73 , "contactSyncUps
000046C0 65 6C 6C 53 68 6F 77 6E 22 3A 66 61 6C 73 65 7D ellShown": false}
000046D0 01 1B 5F 68 74 74 70 73 3A 2F 2F 64 69 73 63 6F _https://disco
000046E0 72 64 2E 63 6F 6D 00 01 74 6F 6B 65 6E 3E 01 22 rd.com token> "
000046F0 4F 54 59 7A 4F 44 67 32 4D 7A 67 32 4E 54 45 34 OTYzODg2Mzg2NTE4
00004700 4D 7A 41 79 4E 7A 49 77 2E 59 6C 63 6E 66 41 2E MzAyNzIw.YlcnfA.
00004710 6C 5F 35 42 78 72 50 2D 58 35 69 49 2D 43 4D 48 l_5BxRP-X5iI-CMH
00004720 47 74 61 77 64 6F 64 6D 69 6F 6D 22 01 1C 5F 68 GtawdodmiOM" _h
00004730 74 74 70 73 3A 2F 2F 64 69 73 63 6F 72 64 2E 63 ttps://discord.c
00004740 6F 6D 00 01 74 6F 6B 65 6E 73 55 01 7B 22 39 36 om tokensU {"96
00004750 33 38 38 36 33 38 36 35 31 38 33 30 32 37 32 30 386386518302720
00004760 22 3A 22 4F 54 59 7A 4F 44 67 32 4D 7A 67 32 4E ": "OTYzODg2Mzg2N
00004770 54 45 34 4D 7A 41 79 4E 7A 49 77 2E 59 6C 63 6E TE4MzAyNzIw.Ylcn
00004780 66 41 2E 6C 5F 35 42 78 72 50 2D 58 35 69 49 2D fA.l_5BxRP-X5iI-

```

Fig. 2. Account settings artifacts recovered from local storage.

```

00003410 7D 0F 8C B3 2B 5B 02 01 A3 01 00 00 00 00 00 00 ) @+{ _E
00003420 06 00 00 00 01 18 4D 45 54 41 3A 68 74 74 70 73 META:https
00003430 3A 2F 2F 64 69 73 63 6F 72 64 2E 63 6F 6D 0D 08 ://discord.com
00003440 84 DA AE F6 D9 FD CE 17 10 DE 81 18 01 27 5F 68 ,0x00y1 _h
00003450 74 74 70 73 3A 2F 2F 64 69 73 63 6F 72 64 2E 63 https://discord.c
00003460 6F 6D 00 01 53 74 72 65 61 6D 65 72 4D 6F 64 65 om StreamerMode
00003470 53 74 6F 72 65 40 01 7B 22 5F 73 74 61 74 65 22 Store {"_state"
00003480 3A 7B 22 39 36 33 38 38 36 33 38 36 35 31 38 33 :{"963863865183
00003490 30 32 37 32 30 22 3A 7B 22 65 6E 61 62 6C 65 64 02720": {"enabled
000034A0 22 3A 74 72 75 65 7D 7D 2C 22 5F 76 65 72 73 69 ":true},"_versi
000034B0 6F 6E 22 3A 31 7D 01 35 5F 68 74 74 70 73 3A 2F on":1} _5_https://

```

Fig. 3. Streamer mode settings artifacts recovered from local storage.

```

{"id": "96386386518302720", "username": "kimkarter", "avatar": null,
"avatar_decoration": null, "discriminator": "9111", "public_flags": 0, "flags": 0,
"banner": null, "banner_color": null, "accent_color": null, "bio": "", "pronouns": "",
"locale": "en-US", "nsfw_allowed": true, "mfa_enabled": false, "analytics_token":
"OTYzODg2Mzg2NTE4MzAyNzIw.sVeJC05GiCsFqI8s47p5pHSAZ2Y", "email":
"kimkarter074@gmail.com", "verified": true, "phone": null}

```

Fig. 4. Account settings artifacts recovered from cache.

```

{"user": {"id": "964551652159918090", "username": "alphen", "avatar": null,
"avatar_decoration": null, "discriminator": "5010", "public_flags": 0, "flags":
0, "banner": null, "banner_color": null, "accent_color": null, "bio": ""},
"connected_accounts": [{"type": "twitch", "id": "788171937", "name":
"alphen178", "verified": true}], "premium_since": null, "premium_guild_since":
null, "mutual_guilds": [{"id": "965385143298691152", "nick": null}]}

```

Fig. 5. Account settings artifacts recovered from cache.

```

{"id": "963876655909179392", "type": 1, "nickname": null, "user": {"id":
"963876655909179392", "username": "jdoe67160", "avatar": null,
"discriminator": "7342", "public_flags": 0}, {"id": "964551652159918090",
"type": 3, "nickname": null, "user": {"id": "964551652159918090", "username":
"alphen", "avatar": null, "discriminator": "5010", "public_flags": 0}]}

```

Fig. 6. Account relationship artifacts recovered from cache.

```

{locale: "en-US", status: "online", theme: "dark", restrictedGuilds: [],...}
  contactSyncUpsellShown: false
  customStatus: {text: "Its Kim!", expiresAt: "2022-04-24T05:00:00+00:00"}
    expiresAt: "2022-04-24T05:00:00+00:00"
    text: "Its Kim!"
  defaultGuildsRestricted: false
  friendDiscoveryFlags: 0
  friendSourceFlags: {all: true}
    all: true
  guildFolders: []
  guildPositions: []
  locale: "en-US"
  restrictedGuilds: []
  status: "online"
  theme: "dark"
  timezoneOffset: 300

```

Fig. 7. Account setting artifacts from Google Developer tools.

in Fig. 5. Connections are vital pieces of artifacts that forensic investigators can use to piece together which accounts were connected alongside any other supplementary information, such as usernames used for other applications associated with the same person.

Moreover, Discord creates additional cache files with each initiated connection request, such as the one shown in Fig. 12. This figure shows a file named `authorize.json` with the URL "<https://Discord.com/api/v9/connections/facebook/authorize>", which is a Facebook authorization request. Other `authorize.json` files will be present for each initiated connection, such as "<https://Discord.com/api/v9/connections/twitch/authorize>" for the twitch application and "<https://Discord.com/api/v9/connections/youtube/authorize>" for the youtube application. Lastly, Fig. 13 shows a picture of the History file URLs table from Google Chrome data which lists out the previously visited links and the time the URLs were visited. The table shows that Facebook and YouTube accounts were logged into, which may suggest that these accounts were connected from Discord. This source can also give a forensic examiner a trail of events and the website links visited to follow and investigate.

4.5. Desktop notifications

One of the preliminary settings for the accounts included turning on desktop notifications. Some of the artifacts recovered with regards to desktop notifications include type, sound settings, taskbar flash, number of notifications, and Guild ID as shown in Table 6.

Figs. 14 and 15 display artifacts from the `local storage` files of Google Chrome that relate to desktop notifications. Fig. 14 presents a snippet of the file with artifacts such as the number of message notifications, whether they were collapsed or not, and the channel ID that the notifications were for. Fig. 15 shows a snippet with artifacts like whether notification sounds are enabled or disabled, unread badges, and taskbar flash settings. Lastly, Fig. 16 depicts desktop notification artifacts from the Chrome developer tools. Some artifacts recovered, such as desktop type, sound settings, unread badge, and taskbar flash, are similar to what was recovered from the local storage file.

4.6. Muting channels

During the use case scenarios, we also tried muting some

Table 3
Account login and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Account Login	Login Time Token	C C

channels in the server to find out if any related artifacts were found. Some artifacts, such as muted server ID, selected time window, and end time, were found from the `local storage.log` file as listed in Table 7.

Fig. 17 shows the local storage file snippet which contains artifacts related to the activity of muting channels. The most important artifacts recovered were the server ID and the time window for muting the channel. Fig. 18 shows the artifacts related to muting a channel from Google Chrome developer tools. Both sources reveal similar artifacts for the use case of muting a channel in Discord.

4.7. Sending text messages

One of the most important aspects of a social media application is the feature to send and receive messages. Some artifacts from this treatment activity included Message ID, content, Author ID, and timestamp as listed in Table 8. This information provides a treasure trove of evidence during a forensic investigation as it can paint a clear picture of the conversations and the time they occurred.

Fig. 19 shows a snippet of a cache file with artifacts relating to sending text messages. Here we can see in plain sight the message that was sent, the author of the message, the ID of the channel it was sent on, and the timestamp of the message. This means that an investigator can uncover the message, who sent it, where it was sent, and when it was sent. If a message has been pinned, then the pinned property will be "true," as shown in Fig. 20. Similarly, if a message is edited, then an edited timestamp will be shown in the cache file with the time the message was updated, as shown in Fig. 21. Fig. 22 displays a screenshot of the pinned and edited messages the cache files depicted from the use case scenario to corroborate the artifacts we found. Finally, Discord allows its user to send spoiler messages (hidden messages). Spoiler tags are used to send hidden messages to users. Spoiler tags can be spotted with two vertical bars before and after the spoiler message, as shown in Fig. 23.

Message reply artifacts were also found in the cache. Some of the important artifacts include the message reference (the message that was replied to), timestamps, and content of the reply that relate to message replies, as shown in Fig. 24.

4.8. Emoji reaction to messages

Throughout the use case scenarios, many treatment activities employed emojis to react to messages. The results revealed artifacts based on this treatment activity. Table 9 lists out some of the artifacts recovered regarding emoji reactions, including the number of reactions, emoji used for the reaction, timestamp, and author artifacts.

Fig. 25 shows the cache file snippet with artifacts related to an emoji reaction. The properties of the emoji reaction are listed under the "reaction" key, which lists out the emoji's name and the count. The name of the emoji is in the form of JSON-encoded Unicode characters, which, when converted to UTF-8 encoded, will reveal a human-understandable emoji.

4.9. Using emojis and stickers

The usage of emojis and stickers can reveal a lot of psychological information about the person. Additionally, the user's mood during the conversation can also be inferred through emojis; thus, it becomes a vital source of intelligence during forensic investigations. Table 10 displays the artifacts found from the usage of emojis and stickers in the experiment.

Fig. 26 shows a snippet of the local storage file, which depicts the artifacts related to using emojis on Discord. The primary two

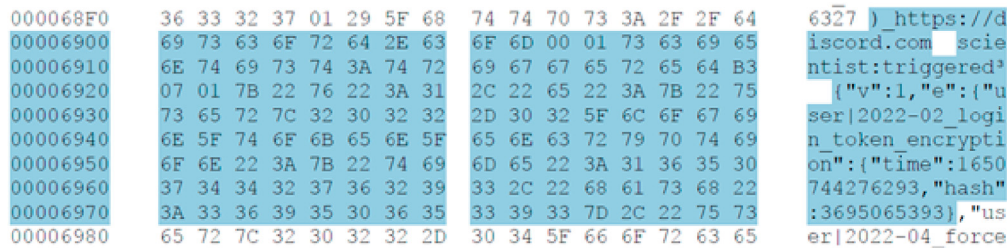


Fig. 8. Account login artifacts recovered from local storage.

Table 4
Adding a payment method and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Adding a payment method	User ID	C
	Brand	C
	Last four digits	C
	Expiry month and year	C
	Billing address	C

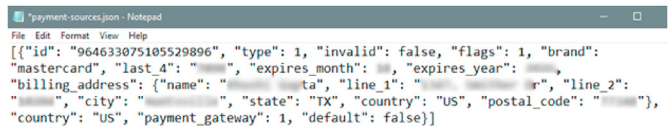


Fig. 9. Account connections artifacts recovered from cache.

Table 5
Account connections and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Connecting 3rd party accounts	Time	C
	Type	C, H
	User ID	C
	Name	C
	Show activity settings	C
	Verification Status	C
	Friend Sync Status	C
	Access token	C



Fig. 10. Account connections artifacts recovered from cache.

artifacts include the key/emoji and the timestamp of when it was used. Adding on to the local storage data, Fig. 27 shows a snippet of the cache file with artifacts related to emoji usage. Similar to how emoji reactions to messages were depicted in JSON encoded Unicode characters, using an emoji puts in the name of the emoji used in the same format under the key “content” as shown in Fig. 27. Apart from the local storage files and the cache files, artifacts on emoji usage can also be found in the Google Chrome developer tools, as shown in Fig. 28. The artifacts found in Chrome developer tools are similar to what was stored in the local storage files, which give information on the key/emoji used and the associated timestamp.

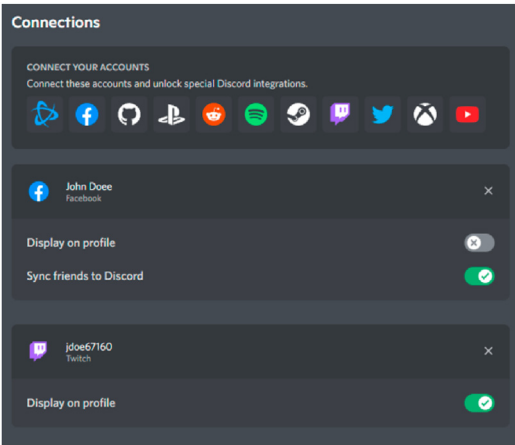


Fig. 11. Account connections in the use case.

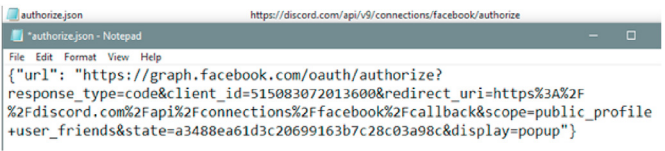


Fig. 12. Account connections artifacts recovered from cache.

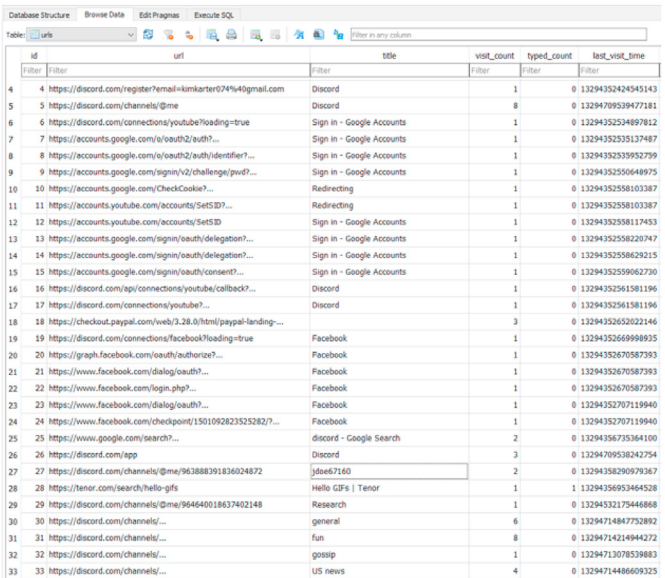


Fig. 13. Account connections artifacts recovered from the history file.

Table 6

Turning on desktop notifications and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Enabling desktop notifications	Type	L, G
	Sound settings	L, G
	Taskbar flash	L, G
	Number of notifications	L
	Channel ID	L

```

000012F0 38 2E 33 34 37 30 30 30 2B 30 30 3A 30 30 22 7D 8.347000+00:00")
00001300 2C 22 6D 65 73 73 61 67 65 5F 6E 6F 74 69 66 69
00001310 63 61 74 69 6F 6E 73 22 3A 33 2C 22 63 6F 6C 6C
00001320 61 70 73 65 64 22 3A 66 61 6C 73 65 2C 22 63 68
00001330 61 6E 6E 65 6C 5F 69 64 22 3A 22 39 36 35 33 38
00001340 35 31 34 33 32 39 38 36 39 31 31 35 35 22 7D 7D
00001350 2C 22 6E 6F 74 69 66 79 5F 68 69 67 68 6C 69 67
00001360 68 74 73 22 3A 30 2C 22 76 65 72 73 69 6F 6E 22
00001370 3A 31 2C 22 6D 75 74 65 5F 63 6F 6E 66 69 67 22

```

Fig. 14. Desktop notifications settings artifacts recovered from local storage.

```

00003260 74 70 73 3A 2F 2F 64 69 73 63 6F 72 64 2E 63 6F tps://discord.co
00003270 6D 0D 08 ED D6 E0 D9 D9 FD CE 17 10 E4 81 18 01 m i0a00yI a
00003280 23 5F 68 74 74 70 73 3A 2F 2F 64 69 73 63 6F 72 # https://discor
00003290 64 2E 63 6F 6D 00 01 6E 6F 74 69 66 69 63 61 74 d.com notificat
000032A0 69 6F 6E 73 9B 01 01 7B 22 5F 73 74 61 74 65 22 ions> {"state"
000032B0 3A 7B 22 64 69 73 61 62 6C 65 64 53 6F 75 6E 64 :{"disabledSound
000032C0 73 22 3A 5B 5D 2C 22 64 69 73 61 62 6C 65 55 6E s":{"disableUn
000032D0 72 65 61 64 42 61 64 74 65 62 3A 66 61 6C 73 65 readBadge":false
000032E0 2C 22 74 61 73 6B 62 61 72 46 6C 61 73 68 22 3A , "taskbarFlash":
000032F0 74 72 75 65 2C 22 74 74 73 54 79 70 65 22 3A 22 true, "ttsType":
00003300 4E 45 56 45 52 2C 22 64 65 73 6B 74 6F 70 54 NEVER, "desktopT
00003310 79 70 65 22 3A 22 41 4C 4C 22 2C 22 64 69 73 61 type": "ALL", "disa
00003320 62 6C 65 41 6C 6C 53 6F 75 6E 64 73 22 3A 74 72 bleAllSounds":tr
00003330 75 65 7D 2C 22 5F 76 65 72 73 69 6F 6E 22 3A 31 ue), "version":1
00003340 7D 01 1B 5F 68 74 74 70 73 3A 2F 2F 64 69 73 63 } _https://disc

```

Fig. 15. Desktop notifications settings artifacts recovered from local storage.

```

{ "_state": { "disabledSounds": [], "disableUnreadBadge": false, "taskbarFlash": true, "ttsType": "NEVER", ... }, ... }
{ "_state": { "disabledSounds": [], "disableUnreadBadge": false, "taskbarFlash": true, "ttsType": "NEVER", ... }, ... }
{ "desktopType": "ALL", "disableAllSounds": true, "disableUnreadBadge": false, "disabledSounds": [], "taskbarFlash": true, "ttsType": "NEVER", ... }
{ "version": 1

```

Fig. 16. Desktop notifications settings artifacts from Google developer tools.**Table 7**

Muting channels and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Muting a channel	Server ID	L, G
	Selected time window	L, G
	End time	L, G

```

00002C60 33 38 35 31 34 33 32 39 38 36 39 31 31 35 32 22 385143258691152"
00002C70 7D 2C 22 36 37 39 38 37 35 39 34 36 35 39 37 30 1), "6798759465970
00002C80 35 36 38 33 22 3A 7B 22 73 75 70 70 72 65 73 56683":{"suppres
00002C90 73 5F 65 76 65 72 79 6E 6E 65 22 3A 66 61 6C 73 s,"suppress_role
00002CA0 65 2C 22 73 75 70 70 72 65 73 73 5F 72 6F 6C 65 e,"suppress_role
00002CB0 73 22 3A 66 61 6C 73 65 2C 22 6D 65 74 65 5F 73 s":false, "mute_s
00002CC0 63 68 65 64 75 6C 65 64 5F 65 76 65 6E 74 73 22 scheduled_events":
00002CD0 3A 66 61 6C 73 65 2C 22 6D 6F 62 69 6C 65 5F 70 :false, "mobile_p
00002CE0 75 73 68 22 3A 74 72 75 65 2C 22 6D 75 74 65 64 65 64 "push":true, "muted
00002CF0 22 3A 74 72 75 65 2C 22 6D 65 73 73 61 67 65 5F notifications":0
00002D00 6E 6F 74 69 66 69 63 61 74 69 6F 6E 73 22 3A 30 , "channel overri
00002D10 2C 22 63 68 61 6E 6E 65 6C 5F 6F 76 65 72 72 69 des":{"_notify
00002D20 64 65 73 22 3A 7B 7D 2C 22 6E 6F 6E 73 22 3A 30 highlights":0, "v
00002D30 68 69 67 68 6C 69 67 68 74 73 22 3A 30 2C 22 76 ersion":6, "mute
00002D40 65 72 73 69 6F 6E 22 3A 36 2C 22 6D 75 74 65 5F config":{"select
00002D50 63 6F 6E 66 69 67 22 3A 7B 22 73 65 6C 65 63 74 ed_time_window":
00002D60 65 64 5F 74 69 6D 65 5F 77 69 6E 64 6F 77 22 3A 3600, "end_time":
00002D70 33 36 30 30 2C 22 65 6E 64 5F 74 69 6D 65 22 3A 32 "2022-04-23T20:2
00002D80 22 32 30 32 32 2D 30 3A 2D 32 33 54 32 30 3A 32 2:48.727000+00:0
00002D90 32 3A 34 38 2E 37 32 37 30 30 2B 30 30 3A 30 0"}, "hide_muted
00002DA0 30 22 7D 2C 22 68 69 64 65 5F 6D 75 74 65 64 5F channels":false,
00002DB0 63 68 61 6E 6E 65 6C 73 22 3A 66 61 6C 73 65 2C "guild_id":"6798
00002DC0 22 67 75 69 6C 64 5F 69 64 22 3A 22 36 37 39 38 75946597056683"}
00002DD0 37 35 39 34 36 35 39 37 30 30 35 36 36 38 33 22 7D

```

Fig. 17. Mute settings artifacts recovered from local storage.

```

{ "_state": { "collapsedGuilds": { "679875946597056683": false, "965385143298691152": false }, "userGuildSettings": { ... } }, ... }
{ "collapsedGuilds": { "679875946597056683": false, "965385143298691152": false }, ... }
{ "679875946597056683": { "suppress_roles": false, "mute_scheduled_events": false, "mobile_push": true, ... } }, ... }
{ "channel_overrides": { "guild_id": "679875946597056683", "hide_muted_channels": false, "message_notifications": 0, "mobile_push": true, "mute_config": { "selected_time_window": 3600, "end_time": "2022-04-23T20:22:48.727000+00:00", "selected_time_window": 3600, "mute_scheduled_events": false, "muted": true, "notify_highlights": 0, "suppress_everyone": false, "suppress_roles": false, "version": 6 } } }, ... }

```

Fig. 18. Mute settings artifacts recovered from Google developer tools.

Discord has its own sticker packs that users can use, which are stored in the cache. Fig. 29 shows all the artifacts recovered from the cache files regarding the usage of stickers. The artifacts include the sticker ID and name, which can be used to further search for the type of sticker using the sticker pack information stored in the cache.

4.10. Uploading media on chats

Media attachments are a common functionality of social media applications often used by users to send digital media to other users. Artifacts about media usage can yield a lot of useful information for forensic investigations. Table 11 presents the artifacts that were recovered from using images, videos, and documents while chatting.

Figs. 30–32 reveal a snippet of the cache file with artifacts related to using media while chatting. These artifacts are very similar to sending text messages with the difference of the attachments key, which declares if the attachment is an image shown in Fig. 30, a video as shown in Fig. 31, or a document as shown in Fig. 32. Other nested key-value pairs include ID, size, proxy URL, height, and width for images and videos.

4.11. Using GIFs and web links

Using GIFs and Weblinks is one of the everyday actions among users on social media platforms. Table 12 shows the artifacts that were recovered from the usage of GIFs and weblinks during the experiment. Some of the artifacts include thumbnails, video URLs, timestamps, and embedded information such as type, URL, title, and provider.

The artifacts related to using a GIF and web URLs in chats are stored under the “embeds” key as depicted in Figs. 33 and 34. The embed subkey “type” sheds more light on whether a GIF (gifv) or a web link (link) is embedded. Other nested artifacts of the embed key include the type, URL, provider, thumbnail URL, proxy URL, height, and width.

4.12. Downloading attachments

The usage of media in messaging applications also involves downloading the media. Some of the artifacts recovered through this activity include tab URL, eTag, MIME type, location of the file in the device, total bytes, and start time, as shown in Table 13. All these artifacts can be recovered from the history file stored by Google Chrome, as shown in Fig. 35. The downloaded files can also be found in the cache under the path “https://cdn.Discordapp.com/attachments/” in case of video files or “https://media.Discordapp.net/attachments/” in case of image files.

4.13. Audio and video calls

Audio and video calls are essential features of most social networking applications. It can yield lots of vital information for a

Table 8
Sending text messages and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Sending a chat	Message ID	C
	Content	C
	Channel ID	C
	Author ID, Username and Discriminator	C
	Timestamp	C
	Pinned settings	C
	Edited timestamp	C
	Spoiler message	C
	Message references (Channel and Message ID)	C

```

1 discordapp.com/external/4d8f9x3701cUDCNH3CqHmHaY2Q?B=0&K=qz3WNYR/https://c.tenor.com/QgdgJ6UwAAAAAC/hello
2 hi.gif", "width": 498, "height": 400, "mentions": [], "pinned": true,
3 "mention_everyone": false, "tts": false, "timestamp": "2022-04-13T20:56:01.624000+00:00", "edited_timestamp":
4 null, "flags": 0, "components": [], "reactions": [{"emoji": "id": "ud83dude929", "count": 1,
5 "true": true}], "channel_id": "963889386518302720", "username": "kinkarater", "avatar": null, "discriminator": "9111",
6 "public_flags": 0, "attachments": [], "embeds": [], "mentions": [], "mention_roles": [], "pinned": false,
7 "mention_everyone": false, "tts": false, "timestamp": "2022-04-13T20:56:29.230000+00:00", "edited_timestamp":
8 null, "flags": 0, "components": [], "reactions": [{"emoji": "id": "ud83dude929", "count": 1,
9 "true": true}], "channel_id": "963889386518302720", "username": "kinkarater", "avatar": null, "discriminator": "9111",
10 "channel_id": "963889386518302720", "author": {"id": "963889386518302720", "username": "kinkarater", "avatar":

```

Fig. 19. Text message artifacts recovered from cache

[illegible]

Fig. 20. Pinned message artifacts recovered from cache.

[illegible]

Fig. 21. Edited message artifacts recovered from cache.

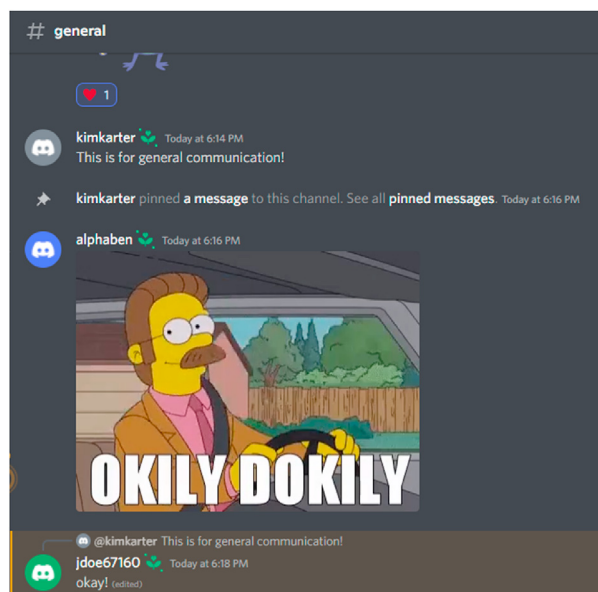


Fig. 22. Messages sent in the use case.

```
{
  "edited_timestamp": null,
  "flags": 0,
  "components": [
    {
      "id": "7490546607692186381",
      "format_type": "name",
      "name": "Wave"
    },
    {
      "id": "96394990546628675",
      "type": "e_content",
      "content": "[How are you doing?]",
      "channel_id": "9638839139071487",
      "author": {
        "id": "963876055909179392",
        "username": "jdoe056",
        "avatar": null,
        "avatar_decoration": null,
        "discriminator": "7242",
        "public_flags": 0,
        "attachments": [
          {
            "embeds": [
              {
                "mentions": [
                  {
                    "pinned": false,
                    "mention_everyone": false,
                    "title": false,
                    "timestamp": "2022-04-11T20:54:18.235000+00:00"
                  }
                ],
                "edited_timestamp": null,
                "flags": 0,
                "components": [
                  {
                    "id": "9639408138096769407",
                    "type": "e_content",
                    "content": "Hello",
                    "channel_id": "963883811836024872",
                    "author": {
                      "id": "963876055909179392",

```

Fig. 23. Spoiler message artifacts recovered from cache.

[illegible]

Fig. 24. Message reply artifacts recovered from cache.

forensic investigation, such as the communication chains and relationships between the users. The artifacts recovered from audio and video calls in Discord are highlighted in [Table 14](#). However, some artifacts related to muting the microphone, turning on and switching off video, and sharing the screen were not found.

Fig. 36 lays out a snippet of the cache file with some of the artifacts related to audio calls in Discord. The artifacts, such as the unique IDs of the participants and the ended timestamp, are listed under the call key. The artifacts stored in the local storage file are somewhat different than what was found in the cache files in the sense that the log storage file reveals the server ID in which the audio channel was found, the Unique ID of the voice channel, and the last connected time in Unix time as shown in Fig. 37.

4.14. Creating a server invite

In order to invite users to the server, server invites are created. These server invites include a variety of information on the server, and therefore, it can prove to be very useful for forensic investigations. Artifacts recovered from the creation of server invites are highlighted in [Table 15](#).

Cache files containing event invite artifacts start with the URL `"https://Discord.com/api/v9/invites/".` Fig. 38 shows one such file with the server invite artifacts. Critical pieces of information include the invite code, expiry date, server ID, name, description (if any), details on the inviter, approximate member count, and the approximate presence count. All the information presented in Fig. 38 can be cross-checked with the actual information from the Discord application shown in Fig. 39. These artifacts can have a significant impact on the investigation as they can reveal critical information on the interactions between people, the number of people in a group, etc.

Table 9

Using emojis to react to a message and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Emoji reaction to a message	Message ID Content Channel ID Author ID, Username and Discriminator Timestamp Emoji reaction	C C C C C C

```

0, "components": []], {"id": "964640253178095680", "type": 0, "content": "Welcome", "channel_id":
"964640018637402148", "author": {"id": "96387665909179392", "username": "jdoe67100", "avatar":
null, "avatar_decoration": null, "discriminator": "73327", "public_flags": 0}, "attachments": [],
"embeds": [], "mentions": [], "mention_roles": [], "pinned": false, "mention_everyone": false,
"tts": false, "timestamp": "2022-04-15T21:35:58.507000+00:00", "edited_timestamp": null, "flags":
0, "components": [], "reactions": [{"emoji": {"id": null, "name": "\u2764\ufe0f"}, "count": 1,
"me": false}], {"id": "964640155229118525", "type": 4, "content": "Research", "channel_id":

```

Fig. 25. Emoji reaction artifacts recovered from cache.**Table 10**

Using Emojis and stickers while chatting and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Using emojis and stickers in chat messages	Message ID Content Channel ID Author ID, Username and Discriminator Timestamp Sticker item (Sticker ID, format type)	C L, C, G C C L, C, G C

```

00000120 3A 2F 2F 64 69 73 63 6F 72 64 2E 63 6F 6D 0D 08 //discord.com
00000130 CD AB FD 94 E9 E9 CE 17 10 B6 5B 17 01 22 5F 68 key=ee1 &_h
00000140 74 74 70 73 3A 2F 2F 6A 69 73 63 6F 72 64 2E 63 https://discord.c
00000150 6F 6D 00 01 45 6D 6F 6A 69 53 74 6F 72 65 56 32 om EmojiStoreV2
00000160 83 01 01 7B 22 5F 73 74 61 74 65 22 3A 7B 22 70 {"_state": "p
00000170 65 6E 64 69 6E 67 55 73 61 67 65 73 22 3A 5B 7B endingUsagees":{"
00000180 22 6B 65 79 22 3A 22 67 72 69 6E 6E 69 6E 67 22 "key": "grinning",
00000190 2C 22 74 69 6D 65 73 74 61 6D 70 22 3A 31 36 34 "timestamp": 164
000001A0 39 38 38 33 33 30 36 37 36 30 7D 2C 7B 22 6B 65 983306760), ("ke
000001B0 75 22 3A 22 73 6D 69 6C 65 25 2C 22 74 69 6D 65 y": "smile", "time
000001C0 73 74 61 6D 70 22 3A 31 36 30 30 35 38 36 38 stamp": 165005868
000001D0 35 33 34 39 7D 6D 7D 2C 22 5F 76 65 72 73 69 6F 5349}], "_versio
000001E0 6E 22 3A 30 7D 65 47 6D 36 0D 01 01 B5 00 00 00 n": 0, "versio
000001F0 00 00 00 00 02 00 00 00 01 18 4D 45 54 41 3A 6B MEArh
00000200 74 74 70 73 3A 2F 2F 6A 69 73 63 6F 72 64 2E 63 https://discord.c
00000210 6F 6D 00 0E AE A2 AD 82 EA E9 CE 17 10 E6 8B 17 om 80-ee1 &_h
00000220 01 22 5F 68 74 74 70 73 3A 2F 2F 6A 69 73 63 6F https://discor
00000230 72 64 2E 63 6F 6D 00 01 45 6D 6F 6A 69 53 74 6F rd.com EmojiSto
00000240 72 65 56 32 B3 01 01 7B 22 5F 73 74 61 74 65 22 reV2 {"_state":
00000250 3A 7B 22 70 65 6E 6A 69 6E 67 55 73 61 67 65 73 {"_pendingUsagees
00000260 22 3A 5B 7B 22 6B 65 79 22 3A 22 67 72 69 6E 6E "":{"key": "grinn
00000270 69 6E 67 22 2C 74 69 6D 65 73 74 61 6D 70 22 6E 67 22 ing", "timestamp":
00000280 3A 31 36 34 39 38 38 33 33 30 36 37 36 30 7D 2C 164983306760),
00000290 7B 22 6B 65 79 22 3A 22 73 6D 69 6C 65 22 2C 22 ("key": "smile", "
000002A0 74 69 6D 65 73 74 61 6D 70 22 3A 31 36 35 30 30 timestamp": 16500
000002B0 35 38 36 38 35 33 34 39 5D 2F 72 62 6B 65 79 22 58685349), ("key":
000002C0 3A 22 75 70 73 69 6A 65 7F 64 67 77 6E 22 2C 22 "upside_down", "
000002D0 74 69 6D 65 73 74 61 6D 70 22 3A 31 36 35 30 30 timestamp": 16500
000002E0 35 39 31 38 33 31 31 35 7D 5D 7D 2C 22 5F 76 65 59183150}], "_ve
000002F0 72 73 69 6F 6E 22 3A 30 7D A3 73 01 9B 4A 01 01 sion": 0})&s NJ
00000300 E7 00 00 00 00 00 00 00 02 00 00 00 01 18 4D 45 ME

```

Fig. 26. Emoji usage artifacts recovered from local storage.

```

null, "flags": 0, "components": []], {"id": "964640784836079686", "type": 0,
"content": "\u2728\u2728", "channel_id": "964640018637402148", "author": {"id":
"963886386518302720", "username": "kimkarter", "avatar": null, "avatar_decoration":
null, "discriminator": "9111", "public_flags": 0}, "attachments": [], "embeds": [],
"mentions": [], "mention_roles": [], "pinned": false, "mention_everyone": false,
"tts": false, "timestamp": "2022-04-15T21:38:05.264000+00:00", "edited_timestamp":
null, "flags": 0, "components": []], {"id": "964640594335002684", "type": 19,

```

Fig. 27. Emoji Usage artifacts recovered from cache.

4.15. Creating a server event invite

Server event invites are created to host an event on the server. Similar to server invites, they entail loads of useful information for

```

v(_state: {}, _version: 0)
v(_state: {}, _version: 0)
v(pendingUsagees: [{"key": "grinning", timestamp: 164983306760}, {"key": "smile", timestamp: 1650058685349}],
> 0: {"key": "grinning", timestamp: 164983306760}
> 1: {"key": "smile", timestamp: 1650058685349}
> 2: {"key": "upside_down", timestamp: 1650059183150}
> 3: {"key": "heart_eyes", timestamp: 1650236704701}
> 4: {"key": "heart", timestamp: 1650236734355}
> 5: {"key": "joy", timestamp: 1650237842500}
> 6: {"key": "joy", timestamp: 1650237842500}
> 7: {"key": "rofl", timestamp: 1650237842500}
> 8: {"key": "rofl", timestamp: 1650237842500}
> 9: {"key": "smiley", timestamp: 1650239221496}
_version: 0

```

Fig. 28. Emoji usage artifacts recovered from Google developer tools.

```

13T20:55:04.835000+00:00", "edited_timestamp": null, "flags": 0, "components": [],
{"id": "963905055780063307", "type": 0, "content": "", "channel_id":
"963886386518302720", "author": {"id": "963886386518302720", "username": "kimkarter",
"avatar": null, "avatar_decoration": null, "discriminator": "9111", "public_flags":
0}, "attachments": [], "embeds": [], "mentions": [], "mention_roles": [], "pinned":
false, "mention_everyone": false, "tts": false, "timestamp": "2022-04-
13T20:54:33.788000+00:00", "edited_timestamp": null, "flags": 0, "components": [],
"sticker_items": [{"id": "749054660769218631", "format_type": 3, "name": "Wave"}]],
{"id": "963904990546628629", "type": 0, "content": "[[How are you?]]", "channel_id":

```

Fig. 29. Sticker usage artifacts recovered from cache.

forensic examiners. In this experiment, numerous server event invite artifacts were recovered from the cache as listed in Table 16.

Fig. 40 depicts the cache file with artifacts about a server event invite. As shown in the figure, several pieces of useful information are listed, such as the event invite code, expiry date and time, the server details where the event is hosted, the inviter details, and details on the scheduled event (ID, name, description, scheduled start time and end time and the RSVP users). Most of the artifacts presented in the figure can be verified using Fig. 41, which presents a screenshot of the event invite from the Discord platform.

4.16. Server log information

A server log is maintained by Discord that shows the activities happening on the server. This log can only be viewed by the server administrator and contains vital information about the messages, channels, threads, and calls on the server. Table 17 lists the artifacts recovered with regard to the server audit log maintained by Discord.

All the activities happening on the server can also be viewed in the cache file in the form of artifacts recovered from Discord, as shown in Fig. 42. The log entries are listed with additional information such as user ID, target ID, and the changes made. This information can be corroborated using Fig. 43, which displays a screenshot of the audit log from the Discord interface.

4.17. Creation of threads

Discord offers a feature of creating a thread under a channel from a specific message to differentiate the conversation regarding that message from the rest of the conversation in the channel. This can be a helpful feature for malicious users to converse about their plans and actions separately. Artifacts about the creation of threads

Table 11
Media usage while chatting and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Using images, videos and documents in the chat	Message ID Type Channel ID Author ID, Username and Discriminator Attachments (ID, filename, URL, type, size) Timestamp	C C, C C C C

```
{
  "components": [{}],
  "id": "96390580898599486504",
  "type": 0,
  "content": "Assignment",
  "channel_id": "96388391836024872",
  "author": {
    "id": "963876655909179392",
    "username": "jdoe67160",
    "avatar": null,
    "avatar_decoration": null,
    "discriminator": "7342",
    "public_flags": 0,
    "attachments": [{}],
    "id": "963905808599486504",
    "filename": "1.png",
    "size": 64755,
    "url": "https://cdn.discordapp.com/attachments/96388391836024872/963905808599486504/1.png",
    "proxy_url": "https://media.discordapp.net/attachments/96388391836024872/963905808599486504/1.png",
    "width": 804,
    "height": 732,
    "content_type": "image/png",
    "embeds": [],
    "mentions": [],
    "mention_roles": [],
    "pinned": false,
    "mention_everyone": false,
    "tts": false,
    "timestamp": "2022-04-13T20:57:33.367000+00:00",
    "edited_timestamp": null,
    "flags": 0,
    "components": [{}],
    "id": "963905601337978940",
    "type": 6,
    "content": "",
    "channel_id":

```

Fig. 30. Image messages artifacts recovered from cache.

```
null,
"name": "\ud83d\ude43",
"count": 1,
"me": true},
{"id": "964642415644385311",
"type": 0,
"content": "",
"channel_id": "964640018637402148",
"author": {
  "id": "964551652159918090",
  "username": "alphen",
  "avatar": null,
  "avatar_decoration": null,
  "discriminator": "5010",
  "public_flags": 0,
  "attachments": [{}],
  "id": "964642414574829578",
  "filename": "steps.mp4",
  "size": 5886572,
  "url": "https://cdn.discordapp.com/attachments/964640018637402148/964642414574829578/steps.mp4",
  "proxy_url": "https://media.discordapp.net/attachments/964640018637402148/964642414574829578/steps.mp4",
  "width": 1920,
  "height": 1080,
  "content_type": "video/mp4",
  "embeds": [],
  "mentions": [],
  "mention_roles": [],
  "pinned": false,
  "mention_everyone": false,
  "tts": false,
  "timestamp": "2022-04-15T21:44:34.079000+00:00",
  "edited_timestamp": null,
  "flags": 0,
  "components": [{}],
  "id": "964640784836079686",
  "type": 0,
  "content":

```

Fig. 31. Video messages artifacts recovered from cache.

```
[{}],
{"id": "964643199870185552",
"type": 0,
"content": "",
"channel_id": "964640018637402148",
"author": {
  "id": "963886386518302720",
  "username": "kinkarter",
  "avatar": null,
  "avatar_decoration": null,
  "discriminator": "9111",
  "public_flags": 0,
  "attachments": [{}],
  "id": "964643198582550668",
  "filename": "Useful websites.docx",
  "size": 12493,
  "url": "https://cdn.discordapp.com/attachments/964640018637402148/964643198582550668/Useful websites.docx",
  "proxy_url": "https://media.discordapp.net/attachments/964640018637402148/964643198582550668/Useful websites.docx",
  "content_type": "application/vnd.openxmlformats-officedocument.wordprocessingml.document",
  "embeds": [],
  "mentions": [],
  "mention_roles": [],
  "pinned": false,
  "mention_everyone": false,
  "tts": false,
  "timestamp": "2022-04-15T21:47:41.053000+00:00",
  "edited_timestamp": null,
  "flags": 0,
  "components": [{}],
  "id": "964642810777202780",
  "type": 6,
  "content":

```

Fig. 32. Document messages artifacts recovered from cache.

can be recovered from the cache. Table 18 displays the artifacts recovered from the creation of threads.

Fig. 44 depicts the cache file with the artifacts related to the creation of threads on Discord. The artifacts associated with the creation of threads are listed under the thread key of the message thread was created from. The thread key has several nested key-value pairs, which gives more insight into the various artifacts recovered, such as the ID, the server ID, the last message ID, the owner ID, the name of the thread, and many others. The artifacts

Table 12
Using GIFs and web links while chatting and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Using GIFs and weblinks in chats	Message ID Content Channel ID Author ID, Username and Discriminator Embeds (type, URL, title, provider) Thumbnail and video URL Timestamp	C C, C C C C C

```
{
  "id": "96539030023492629",
  "type": 0,
  "content": "https://tenor.com/view/okilly-dokily-okie-dokie-ok-kay-nerd-gif-11941985",
  "channel_id": "965385143298691153",
  "author": {
    "id": "964551652159918090",
    "username": "alphen",
    "avatar": null,
    "avatar_decoration": null,
    "discriminator": "5010",
    "public_flags": 0,
    "attachments": [{}],
    "embeds": [{}],
    "gifv": "https://tenor.com/view/okilly-dokily-okie-dokie-ok-kay-nerd-gif-11941985",
    "title": "kk",
    "provider": {
      "name": "Tenor",
      "url": "https://tenor.co",
      "thumbnail": "https://media.tenor.com/srbg20LCRO5AAAPo/okilly-dokily-okie-dokie-dokie.png?c=VJfFzG1zY29yZA",
      "proxy_url": "https://images-ext-1.discordapp.net/external/wVH5U7Q1XR6S12h1wJ0BFVJ0rTHeWgYfCrU2Pnr580/X3fCg30VJfFzG1zY29yZA/https://media.tenor.com/srbg20LCRO5AAAPo/okilly-dokily-okie-dokie-dokie.png?c=VJfFzG1zY29yZA",
      "width": 376,
      "height": 286,
      "video": {
        "url": "https://media.tenor.com/srbg20LCRO5AAAPo/okilly-dokily-okie-dokie-dokie.mp4?c=VJfFzG1zY29yZA",
        "width": 376,
        "height": 286,
        "proxy_url": "https://images-ext-2.discordapp.net/external/fdc_Cq6bvsBS3CtHJ5WMyng20y1CMVln9g0Cg0t350/X3fCg30VJfFzG1zY29yZA/https://media.tenor.com/srbg20LCRO5AAAPo/okilly-dokily-okie-dokie-dokie.mp4?c=VJfFzG1zY29yZA",
        "pinned": false,
        "mention_everyone": false,
        "tts": false,
        "timestamp": "2022-04-17T23:16:23.010000+00:00",
        "edited_timestamp": null,
        "flags": 0,
        "components": [{}],
        "id": "965390270432686090",
        "type": 6,
        "content": "",
        "channel_id":

```

Fig. 33. GIF artifacts recovered from cache.

```
[{}],
{"id": "963905424191545385",
"type": 0,
"content": "https://tenor.com/search/hello-gifs",
"channel_id": "96388391836024872",
"author": {
  "id": "963886386518302720",
  "username": "kinkarter",
  "avatar": null,
  "avatar_decoration": null,
  "discriminator": "9111",
  "public_flags": 0,
  "attachments": [{}],
  "embeds": [{}],
  "link": "https://tenor.com/search/hello-gifs",
  "title": "Hello gifs | Tenor",
  "description": "With Tenor, maker of GIF Keyboard, add popular Hello animated GIFs to your conversations. Share the best GIFs now sss",
  "provider": {
    "name": "Tenor",
    "url": "https://c.tenor.com/qdqe30U8mMAAAC/hello-hi-gif",
    "proxy_url": "https://images-ext-1.discordapp.net/external/4hfX6sjo30C1KCR0NSJChwaiy202cK89AmoKsh7NRY/https://c.tenor.com/qdqe30U8mMAAAC/hello-hi-gif",
    "width": 498,
    "height": 498,
    "mentions": [{}],
    "mention_roles": [{}],
    "pinned": true,
    "mention_everyone": false,
    "tts": false,
    "timestamp": "2022-04-13T20:56:01.624000+00:00",
    "edited_timestamp": null,
    "flags": 0,
    "components": [{}],
    "reactions": [{}],
    "emoji": {
      "id": null,
      "name": "\ud83d\ude02",
      "count": 1,
      "me": false},
    "id": "963905250836754452",
    "type": 0,
    "content":

```

Fig. 34. Web URL artifacts recovered from cache.

depicted in Fig. 44 can be cross-checked with the actual use case activity performed in the experiment to verify its accuracy (see Fig. 45).

4.18. Joining a public server

Discord users can join third-party public servers based on their interests. There are several servers aimed at games, books, and so on. Various artifacts such as Invite code, server ID, name, banner, description, features, welcome channels, and many more are retrieved from the cache, as shown in Table 19. All of these artifacts offer a wealth of information on the type of server joined by the user, which can further give critical details such as the user's interests.

As shown in Fig. 46, some of the key pieces of artifacts recovered from the cache related to joining a public server are as follows: the invite code, the server ID, the name of the server, the description, the different channels, and the corresponding IDs under the server, the server splash, and banner image ids.

Table 13
Downloading attachments and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Downloading attachments from chat messages	Start time Total bytes Tab URL eTag MIME-type Location on the device	D D D D D D

start_time	received_bytes	total_bytes	state	end_time	tab_url	etag	last_modified	mime_type	original_mime_type
13294712768186890	5019236	5019236	1	13294712771568233	https://cdn.discordapp.com/attachments/...	"58e5b421450b0c69690b2ff20e005d3"	Sun, 17 Apr 2022 23:44:23 GMT	video/mp4	video/mp4

Fig. 35. File downloads artifacts recovered History file.

Table 14
Audio and video calls and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Audio and video calls	Channel ID Timestamp Ended timestamp Participants ID	D D D D

```
{
  "id": "96464442262337626",
  "type": 3,
  "content": "",
  "channel_id": "964640018637402148",
  "author": {
    "id": "96387665909179392",
    "username": "jdoe67160",
    "avatar": null,
    "avatar_decoration": null,
    "discriminator": "7342",
    "public_flags": 0
  },
  "attachments": [],
  "embeds": [],
  "mentions": [
    {
      "id": "963886386518302720",
      "username": "kimkarter",
      "avatar": null,
      "avatar_decoration": null,
      "discriminator": "9111",
      "public_flags": 0
    },
    {
      "id": "964551652159918090",
      "username": "alphen",
      "avatar": null,
      "avatar_decoration": null,
      "discriminator": "5010",
      "public_flags": 0
    }
  ],
  "mention_roles": [],
  "pinned": false,
  "mention_everyone": false,
  "tts": false,
  "timestamp": "2022-04-15T21:52:32.494000+00:00",
  "edited_timestamp": null,
  "flags": 0,
  "components": [
    {
      "type": 1,
      "label": "Call",
      "style": 1,
      "options": [
        {
          "label": "Participants",
          "value": "96387665909179392",
          "type": 1
        },
        {
          "label": "Ended timestamp",
          "value": "2022-04-15T22:00:24.811000+00:00",
          "type": 1
        }
      ]
    }
  ],
  "id": "964644226103136286",
  "type": 0,
  "content": ""
}
```

Fig. 36. Call artifacts recovered from cache.

```
00000700 18 4D 45 54 41 3A 68 74 74 70 73 3A 2F 2F 64 69 META:https://d
00000710 73 63 6F 72 64 2E 63 6F 6D 0D 08 E2 9A BD 99 EC scord.com 8541
00000720 E9 CE 17 10 B6 8C 17 01 2A 5F 68 74 74 70 73 3A 61 90 * https:
00000730 2F 2F 64 69 73 63 6F 72 64 2E 63 6F 6D 00 01 53 //discord.com 8
00000740 65 6C 65 63 74 65 64 43 68 61 6E 6E 65 6C 53 74 electedChannelId
00000750 6F 72 65 E8 01 01 7B 22 73 65 6C 65 63 74 65 64 oreId ["selected
00000760 43 68 61 6E 6E 65 6C 49 64 22 3A 22 39 36 34 36 ChannelId":"9646
00000770 34 30 30 31 38 36 33 37 34 30 32 31 34 38 22 2C 40018637402148",
00000780 22 73 65 6C 65 63 74 65 64 56 6F 69 63 65 43 68 "selectedVoiceCh
00000790 61 6E 6E 65 6C 49 64 22 3A 22 39 36 34 36 34 30 annelId":"964640
000007A0 30 31 38 36 33 37 34 30 32 31 34 38 22 2C 22 6C 018637402148",
000007B0 61 73 74 43 6F 6E 6E 65 63 74 65 64 54 69 6D 65 astConnectedTime
000007C0 22 3A 31 36 35 30 30 35 39 37 36 38 35 33 32 2C "1650059768532,
000007D0 22 73 65 6C 65 63 74 65 64 43 68 61 6E 6E 65 6C "selectedChannel
```

Fig. 37. Call artifacts recovered from local storage.

Table 15
Creating a server invite and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Creating a server invite	Code Expiry Server ID, name, and icon Channel ID, name and type Inviter ID, Username and Discriminator Approximate member count Approximate presence count	C C C C C C C

```
File Edit Format View Help
with_counts=true&with_expiration=true.json - Notepad

{"code": "2gJRwcbq", "type": 0, "expires_at": "2022-04-24T22:56:48+00:00", "guild": {"id": "965385143298691152", "name": "SH", "splash": null, "banner": null, "description": null, "icon": "93d170bcd1e2c42445bbe6e886fcc8eb", "features": [], "verification_level": 0, "vanity_url_code": null, "premium_subscription_count": 0, "nsfw": false, "nsfw_level": 0}, "channel": {"id": "965385143298691155", "name": "general", "type": 0, "inviter": {"id": "963886386518302720", "username": "kimkarter", "avatar": null, "avatar_decoration": null, "discriminator": "9111", "public_flags": 0}, "approximate_member_count": 3, "approximate_presence_count": 1}
```

Fig. 38. Server invite artifacts recovered from cache.

INVITER	INVITE CODE	USES	EXPIRES
kimkarter#9111 #general	2gJRwcbq	2	06:23:04:36
kimkarter#9111 #gossip	trnumSTw	0	06:23:34:43

Fig. 39. Server invite from the use case.

Table 16
Creating a server event invite and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Creating a server event invite	Code Expiry Server ID, name, & icon Channel ID, name & type Inviter ID, Username & Discriminator Approximate member count Approximate presence count Server event, server, channel, & creator ID Description Scheduled start and end time Privacy	C C C C C C C C C C

4.19. Other artifacts

Other artifacts that can be recovered from the local storage include the last selected voice and text channels and the time they were selected. These artifacts are presented as "selectedVoiceChannelId,"

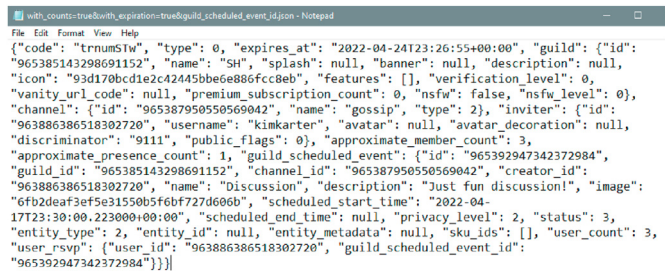


Fig. 40. Server event invite artifacts recovered from cache.

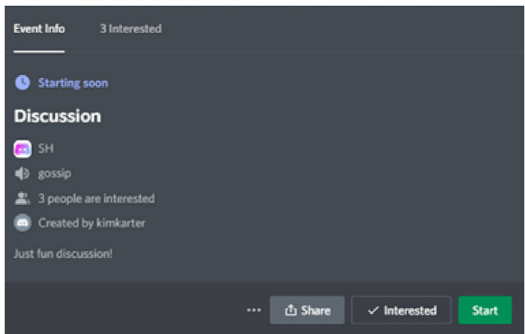


Fig. 41. Server event invite from the use case.

Table 17
Server audit logs and its associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Server log information	Threads	C
	Member IDs	C
	Server scheduled events	C
	Users	C
	Invite code (Channel ID & Inviter ID)	C
	Roles	C
	Channels under the server	C

mostRecentSelectedTextChannelId” with the corresponding channel ID numbers are shown in Fig. 47. These key pieces of artifacts can prove to be very useful in a digital forensics investigation as it gives information on the last used or accessed channel, which can lead the investigator to where a person left off their Discord activity.

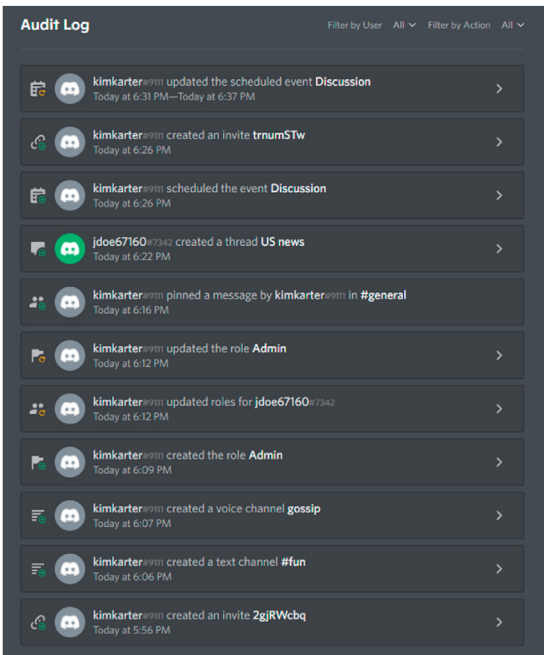


Fig. 43. Server audit logs from the use case.

Table 18
Creation of threads and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Creation of threads under channels in a server	Thread ID	C
	Guild ID	C
	Parent ID	C
	Owner ID	C
	Type	C
	Name	C
	Last message ID	C
	Creation timestamp	C



Fig. 42. Server audit logs artifacts recovered from cache.

```

"edited_timestamp": null, "flags": 0, "components": [], {"id": "965391960695275580", "type": 21,
"content": "", "channel_id": "965391394548121650", "author": {"id": "963876655909179392", "username":
"jdoe67160", "avatar": null, "discriminator": "7342", "public_flags": 0}, "attachments": [], "embeds":
[], "mentions": [], "mention_roles": [], "pinned": false, "mention_everyone": false, "tts": false,
"timestamp": "2022-04-17T23:22:59.545000+00:00", "edited_timestamp": null, "flags": 0, "components": [],
"message_reference": {"channel_id": "965387891196973156", "guild_id": "965385143298691152", "message_id":
"965391394548121650"}, "referenced_message": {"id": "965391394548121650", "type": 0, "content":
"https://www.cnbcm.com/", "channel_id": "965387891196973156", "author": {"id": "963876655909179392",
"username": "jdoe67160", "avatar": null, "discriminator": "7342", "public_flags": 0}, "attachments": [],
"embeds": [{"type": "article", "url": "https://www.cnbcm.com/", "title": "US Top News and Analysis",
"description": "CNBC is the world leader in business news and real-time financial market coverage. Find
fast, actionable information.", "provider": {"name": "CNBC", "thumbnail": {"url":
"https://sc.cnbcm.com/applications/cnbc.com/staticcontent/img/cnbc_logo.gif?v=1524171804&w=1920&h=1080",
"proxy_url": "https://images-ext-1.discordapp.net/external/Aeyds2ZU0QIzy3XpYs88Briz26Zuy08571bV3wXLYE/
%3Fv%3D1524171804%26w%3D1920%26h%3D1080/https://sc.cnbcm.com/applications/cnbc.com/staticcontent/img/cnbc_logo.gif", "width": 1200,
"height": 630}], "mentions": [], "mention_roles": [], "pinned": false, "mention_everyone": false, "tts":
false, "timestamp": "2022-04-17T23:22:59.545000+00:00", "edited_timestamp": null, "flags": 32,
"components": [], "thread": {"id": "965391394548121650", "guild_id": "965385143298691152", "parent_id":
"965387891196973156", "owner_id": "963876655909179392", "type": 11, "name": "US news", "last_message_id":
"965391960695275580"}, "thread_metadata": {"archived": false, "archive_timestamp": "2022-04-
17T23:22:59.533000+00:00", "auto_archive_duration": 1440, "locked": false, "create_timestamp": "2022-04-
17T23:22:59.533000+00:00"}, "message_count": 1, "member_count": 1, "rate_limit_per_user": 0, "flags": 0,
"member_ids_preview": ["963876655909179392"]}]]]]

```

Fig. 44. Thread creation artifacts recovered from cache.

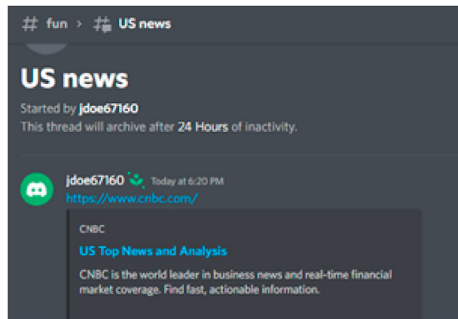


Fig. 45. Thread creation from the use case.

Table 19

Joining public servers and the associated artifacts.

Treatment Activity	Artifacts Found	Storage Locations
Joining a public server	Invite code	C
	Server ID	C
	Name	C
	Banner	C
	Description	C
	Icon	C
	Features	C
	Welcome channels (ID, description)	C
	Approximate member count	C
	Approximate presence count	C
	Recommended server	C

```

with_counts=true&with_expiration=true.json - Notepad
File Edit Format View Help
{"code": "valorantlfg", "type": 0, "expires_at": null, "guild": {"id": "828370043867496531",
"name": "VALORANT LFG", "splash": "0c749989322deecb8f502fda04ae21b7", "banner":
"3a469212a95302603ee5376752ff2569", "description": "The VALORANT Looking for Group Discord
server!", "icon": "a_42b09a30b2b98e25197239bb1e363140", "features": ["PRIVATE_THREADS",
"NEW_THREAD_PERMISSIONS", "MEMBER_VERIFICATION_GATE_ENABLED", "DISCOVERABLE", "COMMUNITY",
"ANIMATED_ICON", "VANITY_URL", "ROLE_ICONS", "SEVEN_DAY_THREAD_ARCHIVE", "THREADS_ENABLED",
"BANNER", "MEMBER_PROFILES", "WELCOME_SCREEN_ENABLED", "INVITE_SPLASH", "THREE_DAY_THREAD_ARCHIVE",
"PREVIEW_ENABLED", "NEWS", "ENABLED_DISCOVERABLE_BEFORE", "ANIMATED_BANNER", "verification_level":
2, "vanity_url_code": "valorantlfg", "premium_subscription_count": 166, "welcome_screen":
{"description": "Welcome to the VALORANT LFG Discord server, in collaboration with Riot Games.",
"welcome_channels": [{"channel_id": "829347062335406080", "description": "Check out our rules",
"emoji_id": "861660999753138226", "emoji_name": "AgentBrimstone"}, {"channel_id":
"855805636847337503", "description": "Read about our LFG system", "emoji_id": "861660977614684213",
"emoji_name": "AgentKay0"}, {"channel_id": "864629778894815273", "description": "Link your VALORANT
account with link", "emoji_id": "861660993850441779", "emoji_name": "AgentCypher"}]}, "nsfw":
false, "nsfw_level": 0}, "channel": {"id": "829347062335406080", "name": "rules", "type": 0},
"approximate_member_count": 372674, "approximate_presence_count": 139418}}

```

Fig. 46. Joining public server artifacts recovered from the cache.

```

00007910 19 03 01 13 02 00 00 00 00 00 00 05 00 00 00 01
00007920 18 4D 45 54 41 3A 68 74 74 70 73 3A 2F 2F 64 69
00007930 73 63 6F 72 64 2E 63 6F 6D 0D 08 94 B6 E9 E0 E6
00007940 FD CE 17 10 A8 80 18 01 2A 5F 68 74 74 70 73 3A
00007950 2F 2F 64 69 73 63 6F 72 64 2E 63 6F 6D 00 01 53
00007960 65 6C 65 63 74 65 64 43 68 61 6E 6E 65 6C 53 74
00007970 6F 72 65 BA 02 01 7B 22 73 65 6C 65 63 74 65 64
00007980 56 6F 69 63 65 43 68 61 6E 6E 65 6C 49 64 22 3A
00007990 6E 75 6C 6C 2C 22 6C 61 73 74 43 6F 6E 6E 65 63
000079A0 74 65 64 54 69 6D 65 22 3A 31 36 35 30 32 34 30
000079B0 36 31 32 30 39 36 2C 22 73 65 6C 65 63 74 65 64
000079C0 43 68 61 6E 6E 65 6C 49 64 73 22 3A 7B 22 39 36
000079D0 35 33 38 35 31 34 33 32 39 38 36 39 31 31 35 32
000079E0 22 3A 22 39 36 35 33 38 35 31 34 33 32 39 38 36
000079F0 39 31 31 35 35 22 6C 22 36 37 39 38 37 35 39 34
00007A00 36 35 39 37 30 35 36 38 33 22 3A 22 38 30 35
00007A10 34 35 30 39 30 32 39 34 30 38 31 31 32 38 35 22
00007A20 7D 2C 22 6D 6F 73 74 52 65 63 65 6E 74 53 65 6C
00007A30 65 63 74 65 64 54 65 78 74 43 68 61 6E 6E 65 6C
00007A40 49 64 73 22 3A 7B 22 39 36 35 33 38 35 31 34 33
00007A50 32 39 38 36 39 31 31 35 32 22 3A 22 39 36 33
00007A60 38 35 31 34 33 32 39 38 36 39 31 31 35 22 2C
00007A70 22 36 37 39 38 37 35 39 34 36 35 39 37 30 35 36
00007A80 36 38 33 22 3A 22 38 30 35 34 35 30 39 30 32 39

```

Fig. 47. Other artifacts recovered from local storage.

5. Discussion

The artifacts created as a result of the treatment contain information about the interactions of users (messages, media, group chats, etc.), time, and configuration setting information. Several relevant artifacts were discovered, including timestamps, emojis used, access tokens, payment information, and many more. As a result, enough artifacts were acquired to allow a digital forensics expert to carve out most of the user's activities.

Despite the fact that the experiment's treatment was designed to isolate user account interactions, more information regarding account setups was discovered. These arrangements were unintended consequences of the treatment and thus can be considered side artifacts. In addition, information on shared activities, such as group chat messages, was discovered. As a result, the involvement of accounts other than the suspect account provides investigators with an additional opportunity to undertake investigations that would otherwise necessitate additional warrants.

The same Discord use case scenarios were also performed on a virtual machine and thus, we analyzed the virtual machine disks for the same artifacts found on the physical machine. Our research shows no difference between physical and virtual machines with respect to recovered Discord artifacts. This knowledge could guide future investigations as digital forensic analysts can confidently use the.vmdk (virtual machine disk file) to recover Discord artifacts if the suspect tried to cover their tracks by using a virtual machine.

6. Conclusion

Online messaging services like Discord are rapidly gaining popularity as more and more people use the internet. This paper presents the findings from a forensic examination of the Google Chrome browser after using the Discord application on it. A quasi-pretest and posttest experiment was performed to evaluate the artifacts left by the Discord application when used on a web browser. Meeting all the research objectives, the results show that a significant amount of information can be recovered from the cache and log files of the browser. When compared to the desktop application of Discord, it was found that Discord employs the same storage mechanism whether used on the desktop or the browser in the Windows environment.

The gathered artifacts displayed potential for further forensic analysis, which will give us a direction for future research in this area. It is feasible to make links between user accounts, their frequency of interactions, and the time periods in which they interact by examining the artifacts identified. These investigations are frequently used to look into collective criminal activities such as organized crime cooperation, drug sales, and collective bullying [Fernández-Planells et al. \(2021\)](#). The emoji content retrieved from the chat messages provides another potential investigative option. Emoji content, usage patterns, and frequency is a concept that researchers employ to assess users' mental states [Marengo et al. \(2017\)](#). Users' emotional responses can be examined for clues about their propensity and eagerness to engage in criminal activity.

Moreover, in the future, we will attempt to analyze the artifacts left by the Discord application in a mobile environment. This will help us correlate the results from existing literature on the desktop environment and our research of the web application used on browsers to the artifacts recovered from a mobile application.

Conflict of interest and authorship confirmation form

Please check the following as appropriate:

All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content, and (c) approval of the final version.

This manuscript has not been submitted to, nor is it under review at, another journal or other publishing venue.

The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript

The following authors have affiliations with organizations with a direct or indirect financial interest in the subject matter discussed in the manuscript:

Appendix

Table 20
Use case scenario 1

Steps	Time	PC1	PC2
1	3:53 pm	jdoe67160 logged in from PC1	
2	3:53 pm		Kimkarter logged in from PC2
3	3:53 pm	A text message "Hello" was sent	
4	3:54 pm	A slash command message was used to send a spoiler message with the text "How are you?"	
5	3:54 pm		The spoiler message is read by clicking on it
6	3:54 pm		Sent a "wave" sticker
7	3:55 pm		A "smiley" emoji is sent and then a text message "Heyyy" is sent
8	3:56 pm		A hyperlink https://tenor.com/search/hello-gifs is sent
9	3:56 pm	A "Laughing" emoji is used to react to the hyperlink sent	
10	3:56 pm	The hyperlink message is pinned	
11	3:56 pm	A text message with a "joy" emoji and "lol" is sent	
12	3:57 pm	A picture named "1.jpg" is sent with the comment "Assignment"	
13	3:57 pm		The message containing the picture is replied to with the text "Thank you!"

Table 21
Use case scenario 2.

Steps	Time	PC1	PC2	PC3
1	4:00 pm	jdoe67160 logged in from PC1		
2	4:29 pm		Kimkarter logged in from PC2	
3	4:29 pm			alphen logged in from PC3
4	4:35 pm	A new group is created named "Research"		
5	4:35 pm	Kimkarter and alphen are added to the group		
6	4:35 pm	A text message "Welcome" is sent		
7	4:36 pm			A "heart" is used to react to the text message "Welcome"
8	4:37 pm		The text message is replied to with the text "Thank you"	
9	4:38 pm		A "smiley" emoji is sent	
10	4:44 pm			A video named "steps.mp4" is sent
11	4:45 pm	The message link containing the video is copied and sent for future		
12	4:46 pm	The message with the link is pinned.		
13	4:46 pm		Reacts to the message with the link with an "upside-down emoji"	
14	4:47 pm		A text file named "Useful_ websites.docx" is sent	
15	4:48 pm	The file "Template.docx" is downloaded.		
16	4:48 pm			The file "Template.docx" is downloaded.
17	4:48 pm		A text message "What is this?" is sent	
18	4:48 pm		The text message "What is this" is deleted	
19	4:50 pm		A text message saying "This is the answer" is sent.	
20	4:50 pm		The text message saying "This is the answer" is edited to read "These are the websites"	
21	4:51 pm	A text message saying "lucky" is sent		
22	4:51 pm			A text message "Thank you!" is sent
23	4:52 pm	The text message that read "lucky" is deleted.		
24	4:52 pm	A video call is requested		
25	4:53 pm		The video call is accepted	The video call is accepted
26	4:53 pm	The video is switched on		
27	4:54 pm			The audio is unmuted
28	4:55 pm	The audio is muted and the video is turned off		
29	4:56 pm		The video chat was ended	The video chat was ended
30	4:57 pm	Started screen share		
31	4:58 pm		Watched the stream share	Watched the stream share
32	4:59 pm	Stopped Streaming		
33	4:59 pm	Chat ended		

References

- Arshad, H., Jantan, A., Omolara, E., 2019. Evidence collection and forensics on social networks: research challenges and directions. *Digit. Invest.* 28, 126–138. Publisher: Elsevier.
- Baggili, I., Breiting, F., 2015. Data sources for advancing cyber forensics: what the social world has to offer. In: 2015 AAAI Spring Symposium Series.
- Brewster, T., 2019. Discord: the \$2 billion gamer's paradise coming to terms with data thieves, child groomers and fbi investigators. <https://www.forbes.com/sites/thomasbrewster/2019/01/29/discord-the-2-billion-gamers-paradise-coming-to-terms-with-data-thieves-child-groomers-and-fbi-investigators/?sh=5d6fd74c3741>. (Accessed 11 September 2022).
- Campbell, S., 2022. Discord statistics 2022: how many people use discord? <https://thesmallbusinessblog.net/discord-statistics/>. (Accessed 11 September 2022).
- Chang, M.S., Yen, C.P., 2020. LinkedIn social media forensics on Windows 10. *Int. J.*

- Netw. Secur. 22, 321–330.
- Cook, T.D., Campbell, D.T., 1979. The design and conduct of true experiments and quasi-experiments in field settings. In: *Reproduced in Part in Research in Organizations: Issues and Controversies*. Goodyear Publishing Company.
- Davis, M., McInnes, B., Ahmed, I., 2022. Forensic investigation of instant messaging services on linux os: discord and slack as case studies. *Forensic Sci. Int.: Digit. Invest.* 42, 301401.
- Dean, B., 2021. How many people use social media in 2022? (65+ statistics). <https://backlinko.com/social-media-users>. (Accessed 11 September 2022).
- Fernández-Planells, A., Orduña-Malea, E., Feixa Pàmpols, C., 2021. Gangs and Social Media: A Systematic Literature Review and an Identification of Future Challenges, Risks and Recommendations. *new media & society*, 1461444821994490.
- Marengo, D., Giannotta, F., Settanni, M., 2017. Assessing personality using emoji: an exploratory study. *Pers. Individ. Differ.* 112, 74–78.
- Moffitt, K., Karabiyik, U., Hutchinson, S., Yoon, Y.H., 2021. Discord forensics: the logs keep growing. In: 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (Cccw). IEEE, pp. 993–999.
- Motylinski, M., MacDermott, A., Iqbal, F., Hussain, M., Aleem, S., 2020. Digital Forensic Acquisition and Analysis of Discord Applications, in: 2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (Ccci). IEEE, pp. 1–7.
- NW3C, 2017. Discord. <https://www.nw3c.org/docs/research/discord.pdf>. (Accessed 11 September 2022).
- Oberlo, 2022. Most Popular Web Browsers in 2022 [sep '22 update] | oberlo. <https://www.oberlo.com/statistics/browser-market-share>. (Accessed 11 September 2022).
- Paligu, F., Varol, C., 2020. Browser forensic investigations of whatsapp web utilizing indexeddb persistent storage. *Future Internet* 12, 184.
- Pandela, T., Riadi, I., 2020. Browser forensics on web-based tiktok applications. *Int. J. Comput. Appl.* 175, 47–52.
- Patel, P., Kannoopatti, K., Shanmugam, B., Azam, S., Yeo, K.C., 2017. A theoretical review of social media usage by cyber-criminals. In: 2017 International Conference on Computer Communication and Informatics (ICCCI). IEEE, pp. 1–6.
- Patterson, D., 2019. Cybercriminals Are Doing Big Business in the Gaming Chat App Discord - Cbs News. <https://www.cbsnews.com/news/cybercriminals-are-doing-big-business-in-the-gaming-chat-app-discord/>. (Accessed 11 September 2022).
- Power, A., 2014. What is social media? *Br. J. Midwifery* 22, 896–897. Publisher: MA Healthcare London.
- Safety, D., 2021. Discord transparency report: Jan — June 2021. <https://discord.com/blog/discord-transparency-report-h1-2021>. (Accessed 11 September 2022).
- Shao, S., Tunc, C., Al-Shawi, A., Hariri, S., 2019. Automated twitter author clustering with unsupervised learning for social media forensics. In: 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA). IEEE, pp. 1–8.
- Sonmez, Y.U., Varol, A., 2019. Legal and technical aspects of web forensics. In: 2019 7th International Symposium on Digital Forensics and Security (ISDFS). IEEE, pp. 1–7.
- UnitedStatesAttorney'sOffice, 2021. Discord user is sentenced to 14 years for distributing child pornography | usao-wdnc | department of justice. <https://www.justice.gov/usao-wdnc/pr/discord-user-sentenced-14-years-distributing-child-pornography>. (Accessed 11 September 2022).