Forensic Investigations of Google Meet and Microsoft Teams – Two Popular Conferencing Tools in the Pandemic

M. A. Hannan Bin Azhar(✉), Jake Timms, and Benjamin Tilley

School of Engineering, Technology and Design, Canterbury Christ Church University, Canterbury, UK
{hannan.azhar,jt472,bt145}@canterbury.ac.uk

Abstract. The Covid-19 pandemic has created unprecedented challenges in the technology age. Previous infrequently used applications were pushed into the spotlight and had to be considered reliable by their users. Applications had to evolve to accommodate the shift in normality to an online world quickly, predominantly for businesses and educational purposes. Video conferencing tools like Zoom, Google Hangouts, Microsoft Teams, and WebEx Meetings can make communication easy, but ease of online communications could also make information easier for cybercriminals to access and to use these tools for malicious purposes. Forensic evaluation of these programs is important, as being able to easily collect evidences against the threat actors will aid investigations considerably. This paper reports how artefacts from two popular video conferencing tools, Microsoft Teams and Google Meet, could be collected and analysed in forensically sound manners. Industry standard cyber forensics tools have been reported to extract artefacts from range of sources, such as memory, network, browsers and registry. The results are intended to verify security and trustworthiness of both applications as an online conferencing tool.

Keywords: Google Meet · MS Teams · Digital forensics · Memory forensics · Network forensics · Video conferencing

1 Introduction

Due to the Covid-19 pandemic, there has been a substantial increase in video calling/conferencing software being utilised. Reference [1] revealed that in March 2020 in the UK, popular applications such as 'Google Hangout/Meet', 'Houseparty', 'Microsoft Teams', and 'Zoom' were downloaded on average 19 times more than in Q4 of 2019. This spike in usage derives solely from the lockdowns and restrictions forced by the pandemic and can be correlated to both an increase in employees working from home and home-schooling. The UK's Office for National Statistics [2] published data that showed 46.6% of people in employment conducted some form of work at home, with 86% of these people doing so because of Covid-19. Similarly, between the months of

P. Gladyshev et al. (Eds.): ICDF2C 2021, LNICST 441, pp. 20–34, 2022.
https://doi.org/10.1007/978-3-031-06365-7_2

May and June 2020, it was discovered that 87% of parents with a child in education had undertaken some form of home-schooling. Whilst video conferencing apps are usually used for work and school meetings, there can also be a darker side to these programs. It might not be as common as work meetings, but video conferencing programs can be used for criminal uses. 'Zoombombing' has become an issue in recent times, according to Wiltshire Police [3]. 'Zoombombing' has been defined as the act of interrupting a zoom call, often with disturbing images of child abuse. This has been allowed by a security flaw in zoom that allows people to join without a password, using a zoom call code that has been posted publicly, such as by pages on Facebook. Whilst there are mitigations for this sort of issue, such as using the "waiting room" feature, and only sending the room code to the people involved, the fact that 'Zoombombing' is happening shows that there is the risk for people to be snooping on calls, or even using them to distribute illegal and disturbing images. As video conferencing has only recently had a boom in popularity, therefore there are not many researches regarding their forensic findings and artefacts. This paper reports how forensic evidence from Microsoft Teams and Google Meet could be collected by forensic examiners, and how these artefacts can be used as evidence. This study will forensically analyse both applications in order to assess their security and provide a detailed review of the features associated with the applications, including any forensic artefacts that could be of interest to investigators or alternatively be used maliciously against users.

The remainder of the paper is organised as follows: Sect. 2 describes existing literature on forensic analysis of similar video conferencing tools and reports the gaps. Section 3 discuses experimental setups for the investigations. Section 4 and 5 reports artefacts recovered for MS Teams and Google Meet respectively. For both applications, various sources of artefacts are reported in detail, including memory, network, registry, etc. Finally, Sect. 6 concludes the paper and give directions to future works.

2 Literature Review

Considering a variety of applications that allow users to make conference calls for social, business, or educational purposes, whether individual or in a group, this section of the literature review intends to uncover and discuss any relevant studies on forensic investigation on similar applications. Acknowledgement of security and privacy issues were first discussed by 'Zoom' in 2020 at the beginning of the first UK lockdown, before the release of 'Zoom 5.0'. The application was understood to have been sending users' device data to 'Facebook' without user permission, wrongfully claiming the application was end-to-end encrypted and unintentionally allowing meeting hosts to track attendees [4]. At this stage also, 'Zoombombing' was at its highest, which is where uninvited guests crashed meetings, including in at least one case displaying pornographic images and shouting profanities [5]. 'Zoom' is not the only application to fall under scrutiny in recent times. 'McAfee' conducted research on Microsoft Teams [6]; with use of more than forty million 'McAfee MVISION Cloud' users worldwide. This research formulated ten prominent security concerns with regards to Microsoft Teams, with issues such as malware being uploaded via Teams, data loss through file sharing in the application and the inclusion of guest users, potentially being inadvertently added to calls where sensitive content may be included.

External security concerns are not the only prevalent issue. However, in 2020, 'Consumer Reports' evaluated the privacy policies of Google Meet, Microsoft Teams and 'Webex' and discovered that these applications may be collecting data whilst in a video-conference to combine with information from data brokers to build consumer profiles and even access video calls in order to train facial recognition systems [7]. Reference [8] conducted a forensic analysis of the 'Zoom' application during the Covid-19 pandemic, when usage statistics were at some of their highest, and discovered that it is possible to find user's data both encrypted and in plain text with information such as chat logs, names, email addresses and passwords. The study involved analysing network traffic and included disk and memory forensics in attempts to obtain notable artefacts that could be of use in an investigation or potentially abused by a malicious user.

The usage of group video calling software has been available for many years. Early applications such as 'Skype', which was released in 2003, quickly utilised better performing networks to allow users to video call. It was from this stage that the importance of security of these applications became prevalent. The Common Vulnerabilities and Exposures or CVE [9], who work alongside some of the biggest software vendors globally is a list of free, publicly disclosed, cybersecurity vulnerabilities found in all forms of software. One such CVE regarding Skype details how attackers could remotely execute arbitrary code on targeted systems by manipulating '.dll' files that Skype loads [10]. Reference [11] analysed 'Skype' on a mobile phone running Android OS 5.5 and discovered that records stored could contain user data and other noteworthy metadata in plain-text format that could be easily accessed by anyone with the physical device.

Aside from Skype and Zoom, existing literature lacks technical investigations of similar popular applications that are widely used and may pose security and privacy threats. This paper contributes to fill this gap by reporting experimental results of forensic investigations of two popular video conferencing applications: MS Teams and Google Meet, both of which have been widely used during the pandemic in a variety of sectors, particularly in business and education.

3 Experimental Setup

To conduct forensic investigation for MS Team, a Windows based virtual machine (VM) was used to capture evidence. With VMs, snapshots can be taken so that any changes can be rolled back if necessary, for instance, when a clean install is needed before any software or files have contaminated the evidence. The choice of Microsoft Windows was based on its popularity in the world with more than three fourths of the global desktop market share [12].

Three test accounts ('is20user1@outlook.com', 'is20user2@outlook.com' and 'is20user3@outlook.com') were created in order to conduct investigations with MS Teams. These accounts were added to join an organisation called 'IS20 Testing'. After the Teams application was installed in the VM, messages were sent from a phone (Samsung Galaxy S8+) with Microsoft Teams installed. Voice calls were used as one of the artefacts. Conducting these calls would show how Teams logs and stores information about them. Testing was done by exploiting the application in the way people would use it on a daily basis.

A similar setup was used to investigate the Google Meet. Four devices were used: two computers (one desktop PC and one laptop) with clean Windows 10 VM installed, one Oracle VM VirtualBox, and finally a Samsung Galaxy S7 mobile phone. In addition to this, a VPN ('NordVPN') was utilised on two of the three devices all times to ensure different IP addresses were assigned to each device during the analysis of network traffic, providing a testing environment mimicking a standard conference call on Google Meet. Scenarios were created and examined, including setups where only the host remained active in the call, one-to-one calls, and group calls with three users. Setups comprised three test accounts created specifically for the experiment, and scenarios were created by connecting meetings using a one-time hyperlink and alternatively through the 'Google Calendar'.

4 Experimental Findings on MS Teams

Microsoft Teams can be used in a web browser, but it is more often used as an installed desktop application. Different types of investigations were performed, including disk, memory and network forensics. Disk forensics examines the artefacts left behind on a device, such as log and cache files. These artefacts could include information such as IP addresses, email addresses, and even messages between users. Since data in use by programs is held in memory, it is likely that there is information in memory could be of use to an investigation. The network is another possible medium in which artefacts might be found. The network traffic can be captured for analysis, such as searching for unencrypted information, and how Teams makes connections, such as whether it uses P2P connections or always connects to Microsoft servers.

4.1 Disk Forensics for MS Teams

During the investigation, the FTK imager was used to capture forensically sound images of both the hard drive. Once the tests were completed and the images captured, the disk drive evidence was placed into the forensic software Autopsy 4.17. Several ingest modules within the Autopsy were run to ensure data integrity and to verify that the evidences have not been tampered with. Also, Autopsy can create a timeline of events and file type identification, which checks for a files MIME type. Checking for MIME types allows for easier finding of files that may not have the correct file extension, such as images and databases. In Autopsy, the disk image was searched for known phrases that have been sent in a channel, as well as using private messages.

Disk forensics revealed that some data such as emails could be found in logs and cache. Windows registry also only held a small number of artefacts, the most useful of which was email addresses. Also, while looking at the Window's registry it was found that MS Teams added the email address to other programs, such as OneDrive and parts of the Windows Security Center.

To explore anti-forensics the program was uninstalled, and search was done to find any artefacts left. This is the usual way some suspects would try to hide their activities. One example of a deleted file recovered after uninstallation was an email address used in a call, as shown in Fig. 1. This shows that Teams can leave behind fairly important information about previous contacts, even after uninstallation.

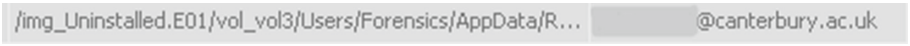


Fig. 1. Recovery of email address after uninstallation of MS Teams.

4.2 Memory Forensics for MS Teams

At several points during the investigation, memory captures were performed. This was due to the fact that memory was constantly changing and would have significantly differed based on the actions of the user, such as sending or receiving a message. For memory forensics, images were searched for relevant files and connections. This included files such as images sent in the channel and by direct message. It also involved network connections that were established. For analysis of memory, 'volatility3' was used, as it is one of the most commonly used tools in memory forensics. Volatility enables searching of any strings open in memory, as well as internet connections, using the command shown in Fig. 2.

```
python3 .\vol.py -f '..\Shared Drive\Images\voiceCall2.mem' windows.netscan
```

Fig. 2. Command in Volatility to show network connections.

Column1	Column2	Column3	Column4	Column5	Column6	Column7	Column8	Column9	Column10
Offset	Proto	LocalAddr	LocalPort	ForeignAddr	ForeignPort	State	PID	Owner	Created
0xe70da7bc2a20	TCPv4	192.168.1.171	62843	52.170.57.27	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:24:04.000000
0xe70da7f4c010	TCPv4	192.168.1.171	62834	13.107.18.11	443	CLOSED	8308	Teams.exe	2021-05-10 15:19:10.000000
0xe70da7f89700	TCPv4	192.168.1.171	62839	51.140.157.153	443	CLOSED	6924	Teams.exe	2021-05-10 15:23:59.000000
0xe70da808c010	TCPv4	192.168.1.171	62635	52.113.199.54	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:13:22.000000
0xe70da8204260	TCPv4	192.168.1.171	62841	52.111.242.2	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:24:03.000000
0xe70da86ed370	TCPv4	192.168.1.171	62849	52.114.128.75	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:24:27.000000
0xe70da8ae62b0	TCPv4	192.168.1.171	62659	52.113.205.20	443	ESTABLISHED	5980	Teams.exe	2021-05-10 15:13:27.000000
0xe70da90e3b10	TCPv4	192.168.1.171	62838	52.114.88.83	443	ESTABLISHED	5980	Teams.exe	2021-05-10 15:23:58.000000
0xe70da9187010	TCPv4	192.168.1.171	62831	52.113.199.99	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:16:33.000000
0xe70da96e3010	TCPv4	192.168.1.171	62837	52.113.194.132	443	ESTABLISHED	8308	Teams.exe	2021-05-10 15:23:21.000000
0xe70da1e7b570	UDPv4	192.168.1.171	50018	*	0		5980	Teams.exe	2021-05-10 15:15:40.000000
0xe70da1e7c380	UDPv4	192.168.1.171	50035	*	0		5980	Teams.exe	2021-05-10 15:15:40.000000
0xe70da22f7e30	UDPv4	0.0.0.0	50995	*	0		5980	Teams.exe	2021-05-10 15:13:27.000000
0xe70da6c95730	UDPv4	0.0.0.0	0	*	0		5980	Teams.exe	2021-05-10 15:13:26.000000
0xe70da9b8ed50	UDPv4	192.168.1.171	50014	*	0		5980	Teams.exe	2021-05-10 15:23:58.000000
0xe70da9dbc700	UDPv4	0.0.0.0	50808	*	0		5980	Teams.exe	2021-05-10 15:13:26.000000
0xe70daa0574e0	UDPv4	0.0.0.0	0	*	0		5980	Teams.exe	2021-05-10 15:13:26.000000
0xe70da22f7e30	UDPv6	::	50995	*	0		5980	Teams.exe	2021-05-10 15:13:27.000000
0xe70da6c95730	UDPv6	::	0	*	0		5980	Teams.exe	2021-05-10 15:13:26.000000
0xe70da9dbc700	UDPv6	::	50808	*	0		5980	Teams.exe	2021-05-10 15:13:26.000000

Fig. 3. Network connections carved from memory. (Color figure online)

Volatility revealed that there were multiple TCP and UDP network connections from the Teams program. The output of volatility was saved into a text file to facilitate easy viewing of the data in Excel. Figure 3 shows the output of a memory capture during a voice call, Teams has a number of TCP connections established with Microsoft servers over port 443, showing the use of encrypted traffic, shown in red. It also shows a number of UDP connections referring to calls made using the App, shown in blue.

The memory image can also be scanned for open files, and strings of information. Using the volatility 'filescan' module, files in use by windows were revealed, and the only files related to Teams were databases, logs, and assets in use by Teams. String analysis revealed that there are email addresses held in RAM. The strings command placed the output in a file, which can then be searched for either specific email

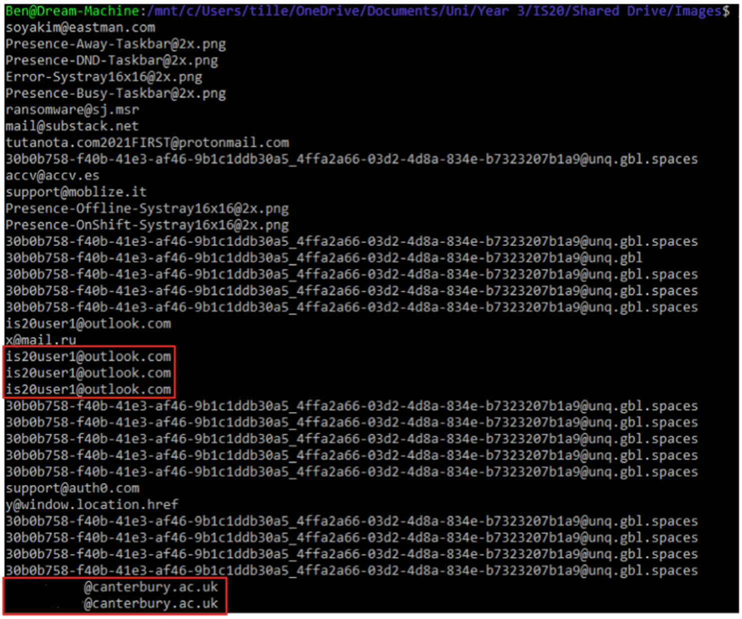


Fig. 4. Strings analysis revealing emails.

addresses, or for an email pattern. Figure 4 shows email addresses related to Teams, such as "is20user1@outlook.com". It also shows email address that were used for testing from outside of the testing organisation, as well as various other emails perhaps used by other programs. Emails that are of use to this investigation are highlighted in red.

The memory was also searched for password strings used for accounts logged in, and none were found. Memory analysis of the captures taken at different times revealed similar artefacts, so only those discovered during a voice call are reported here.

4.3 Network Forensics for MS Teams

By capturing network traffic using Wireshark, investigation of transmitted and received packets can be performed, giving access to any user information, such as log-in details and messages transmitted. Wireshark is a popular network protocol analyser that captures live traffic as it is sent and received on the host machine. The 'Whois' command will also be used to determine ownership of domain names and websites visited.

For network forensic analysis in Teams, a Virtual Machine was created with an IP address 192.168.1.171. When a phone was connected to the same network, it had IP address of 192.168.50.6. As for the phone tested via a 4G mobile network, the IP address was 92.40.175.11. Logging into the Teams client on Windows resulted in a lot of internet traffic. Many of the DNS queries were to obvious places such as login servers owned by Microsoft, however there were a few servers that did not belong to Microsoft. One of these requests is for 'oneclient.sfx.ms', but a 'whois' lookup shows it belongs to 'Akami Technologies', a globally operating caching company, so the use of this server is perhaps not so surprising.

When examining the traffic used for logging in and general communication between the client and the Teams' online service, it was clear that the data exchange was encrypted. The network capture of Fig. 5 and Fig. 6 shows a TLS exchange defining key exchange mechanisms, as well as which cipher suite to be used. Between the client and the server, it was found that Ecliptic Curve Diffie Hellman was used for key exchange, and AES 256 GCM as the encryption. This shows that Microsoft Teams used stronger encryption for data transmission.

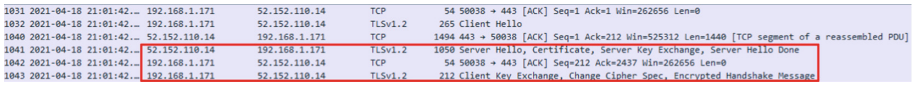


Fig. 5. Wireshark key exchange.

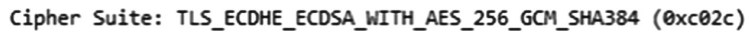


Fig. 6. Wireshark cipher suite.

Further network forensics involved searching the packet capture for phrases that are known to be used in the test organisation, such as "IS20", "Networking", and "general Channel". When Wireshark was used to search in the packet details for those strings, no results were found, indicating that they were not in plain text. Another important network forensics exercise was capturing packets sent during a phone call. Capturing this specific traffic provided a better understanding of how Teams was able to create connections and what type of architecture was used. The main discovery from monitoring the voice call was that Teams utilised a Peer To Peer (P2P) connection between devices. When connecting to a one-on-one call, it was clear that the two clients are talking directly to each other.

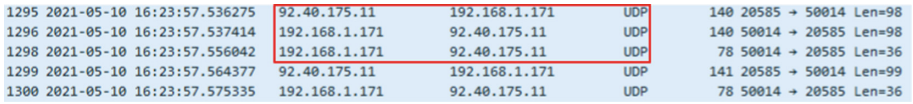


Fig. 7. Voice call P2P conncetion on WAN.

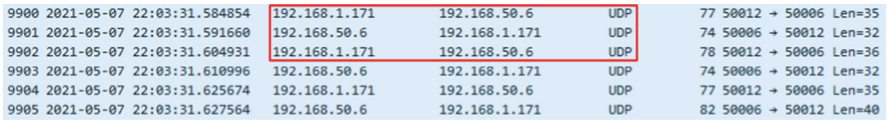


Fig. 8. Voicecall P2P connection on LAN.

As shown in the Fig. 7 and Fig. 8, the two devices have established a UDP stream that appears to circumvent Microsoft's servers. The traffic appears to be streaming directly between devices, both over LAN and WAN. However, when examining the Wireshark

2021-05-10 17:54:34.770423	52.112.97.9	192.168.1.171	UDP	111 plethora(3480) → 50004 Len=69
2021-05-10 17:54:34.794288	192.168.1.171	52.112.97.9	STUN	158 ChannelData TURN Message
2021-05-10 17:54:34.794317	192.168.1.171	52.112.97.9	STUN	158 ChannelData TURN Message
2021-05-10 17:54:34.794892	52.112.97.9	192.168.1.171	UDP	111 plethora(3480) → 50004 Len=69
2021-05-10 17:54:34.810893	52.112.97.9	192.168.1.171	UDP	84 plethora(3480) → 50004 Len=42
2021-05-10 17:54:34.827356	192.168.1.171	52.112.97.9	STUN	93 ChannelData TURN Message

Fig. 9. Wireshark capture of Teams meeting.

Fig. 10. Microsoft server location.

55411 2021-05-07 21:33:52.783032…	192.168.50.6	52.114.132.73	TCP	158 55796 → https(443) [ACK] Seq=16162 Ack=8877 Win=122368 Len=0	
55536 2021-05-07 21:33:53.104935…	192.168.50.6	52.114.76.58	TLSv1.2	482 Application Data	
55549 2021-05-07 21:33:53.121031…	52.114.76.58	192.168.50.6	TLSv1.2	200 Application Data	
55553 2021-05-07 21:33:53.123124…	192.168.50.6	52.114.76.58	TLSv1.2	1400 Application Data	
55584 2021-05-07 21:33:53.139493…	192.168.50.6	52.113.205.254	TLSv1.2	200 Application Data	
55586 2021-05-07 21:33:53.139724…	52.114.76.58	192.168.50.6	TLSv1.2	200 Application Data	
55600 2021-05-07 21:33:53.180742…	52.114.76.58	192.168.50.6	TLSv1.2	317 Application Data	
55604 2021-05-07 21:33:53.182178…	192.168.50.6	52.114.76.58	TCP	158 33492 → https(443) [ACK] Seq=15976 Ack=53694 Win=244096 Len=0	
55608 2021-05-07 21:33:53.182193…	192.168.50.6	52.114.76.58	TCP	158 33492 → https(443) [ACK] Seq=15976 Ack=53853 Win=244096 Len=0	

Fig. 11. Encrypted WiFi traffic.

capture of a meeting (Fig. 9), it appeared to go through a Microsoft server from Amsterdam (Fig. 10), rather than communicating directly. As shown in the Fig. 10, the VM is contacting a Microsoft server, rather than the other clients in the meeting. However, all call traffic was encrypted before being sent through UDP. Additionally, WiFi was monitored for any forensic artefacts, such as transmitting credentials in cleartext. The phone was tested both by logging into Teams and by making an audio call, and all traffic was encrypted between the application and the server, as shown in Fig. 11.

4.4 Registry Forensics for MS Teams

The Windows registry is an important place to check for forensic artefacts, as many of the settings used by Windows and other installed programs are stored in registry hives. This allows forensic investigators to get a good idea of how a computer was set up and used. It also contains history of network interfaces and USB devices, again giving a good idea of how the device was used.

In searching the registry for relevant data before uninstallation, artefacts such as logged in emails, install dates, and locations were found, but no personal data, such as passwords or messages, was located. Figure 12 displays a registry key related to Teams showing the logged in email.

After the uninstallation of the application, used or created emails could be still found in the registry, although these were found when looking for known emails. An interesting

Fig. 12. Registry key showing logged in email.

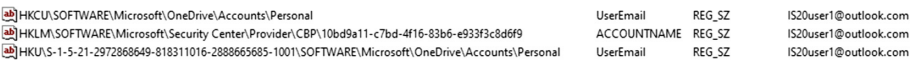


Fig. 13. Remnants in registry.

find was that the email address used by Teams was also linked to OneDrive, as shown in Fig. 13. OneDrive is also a Microsoft product, so it is not too surprising, but it might be worth noting in an investigation.

4.5 Evaluation of Findings for MS Teams

Email addresses and IP addresses are the most common artefacts left by Teams, however a lot of the artefacts that can be recovered from Teams require prior investigation. This means that network activity must already be in the process of being captured, however this cannot be guaranteed. It is similar to memory capture, as this process requires the suspect's computer to be on and running Teams.

Memory analysis revealed some emails and network connections that might be of value. Finding emails might lead to other pieces of evidence in other areas. Network connections in memory may also be useful, however volatility was unable to recover the endpoints of UDP connections. Teams used UDP for activities like calls, so not being able to retrieve that information may negatively impact the investigation. On the other hand, a network monitoring tool such as Wireshark was able to capture the P2P connection during a call. Wireshark captured a lot of information that could be useful for a forensic investigation, such as encryption handshakes and connected IP addresses. Being able to see the IP addresses of devices connected to Teams provides insight into how it works. However, monitoring a suspect's internet traffic is not always possible, and cannot be done after the act has occurred.

5 Experimental Findings on Google Meet

Google Meet is a web-only application, with no options to download the software onto a machine. Additionally, no chat logs, call/meeting history or contact list is available on the application and to set up meetings. It must be simply created instantly with the use of a hyperlink, or created for a future meeting using 'Google Calendar'. Because of the data not being stored locally on the file system, and the application is only being available on a web browser, methods were shifted to attain information surrounding the memory, network, and browser forensics.

5.1 Memory Forensics for Google Meet

We captured the memory on two occasions. The first occurred during an active call between three of the test accounts created for this study, and the second occurred moments

after a call ended. To capture the memory, like before the FTK Imager was used live on the 'Windows 10 Virtual Machine' and saved to an external USB-drive. It creates an image of the memory used on the machine that allows further analysis to be conducted without risking altering the memory being used on the live machine accidentally.

To capture memory with the FTK Imager, the option 'Capture Memory' was used. When this option is selected, a new window appears which requests a destination path to save the memory dump, the filename, whether or not to include the 'pagefile' - a reserved portion of the hard-drive that RAM uses, and finally an option to create an 'AD1 file' - a compressed and hashed version of the memory dump, allowing forensic approval of hash correlation to occur.

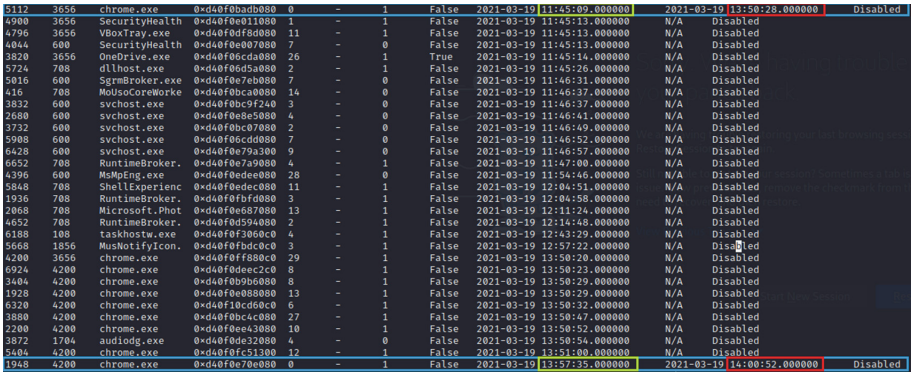


Fig. 14. Memory processes.

After capturing the memory during Google Meet sessions, when analysed in 'Volatility3' with the 'windows.pslist.Pslist' command, a process list was formed. With the example of 'Chrome' being used, this process list outlines multiple "chrome.exe" processes. 'Chrome' creates a separate process for every single web-app, plug-in, tab and extension, explaining the large number of "chrome.exe" processes present. Each process lists a creation time and an exit time, as shown in Fig. 14.

Fig. 15. Volatility Netscan.

Furthermore, when the "windows.netscan" command is used, network connections both live and recently terminated can be recovered. Figure 15 shows two connections with the owner of "chrome.exe". When further analysed, the IP addresses highlighted link to 'Google Cloud' servers located in North America.

5.2 Network Forensics for Google Meet

Like before, again Wireshark was used to capture packet information during Google Meet sessions both a meeting consisting of only an individual test account participating in a call, as well as a call consisting of all three test accounts. On both occurrences, the packet information was the same. The 'shark-fin' icon in Wireshark must be selected once to capture packets and can be paused or stopped at any time (see Fig. 16). The duration of packet capture was approximately thirty seconds on both occasions.

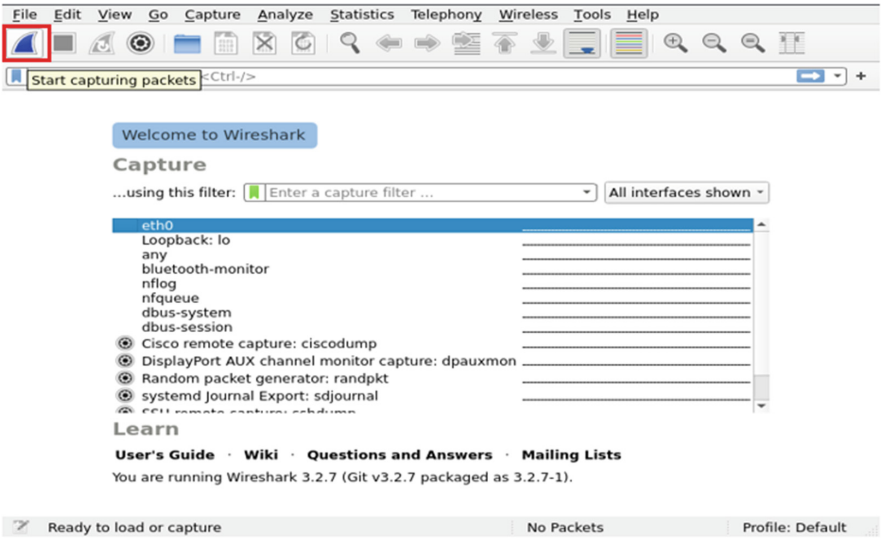


Fig. 16. Wireshark example.

Once packets were captured, a saved log was created for further analysis. The search for keywords or filters to find packets of notability can then be analysed further with a description-style section detailing information about each packet. 'Wireshark' clearly shows the use of TLSv1.2 both when sending and receiving packets (Fig. 17).

No.	Time	Source	Destination	Protocol	Length	Info
135	1.869856939	10.0.2.15	74.125.133.189	TLSv1.2	475	Application Data
136	1.870002767	10.0.2.15	74.125.133.189	TLSv1.2	92	Application Data
137	1.870340482	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=3503 Ack=2898 Win=65535 Len=0
138	1.870340567	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=3503 Ack=2936 Win=65535 Len=0
139	1.903502546	74.125.133.189	10.0.2.15	TLSv1.2	411	Application Data
140	1.904150072	74.125.133.189	10.0.2.15	TLSv1.2	249	Application Data, Application Data, Application Data
141	1.904275930	10.0.2.15	74.125.133.189	TCP	54	51324 → 443 [ACK] Seq=2936 Ack=4055 Win=62780 Len=0
142	1.904314154	10.0.2.15	74.125.133.189	TLSv1.2	93	Application Data
143	1.904548214	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=4055 Ack=2975 Win=65535 Len=0
144	1.921554922	10.0.2.15	74.125.133.189	TLSv1.2	460	Application Data
145	1.921885584	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=4055 Ack=3381 Win=65535 Len=0
146	1.954610029	74.125.133.189	10.0.2.15	TLSv1.2	462	Application Data, Application Data
147	1.995741747	10.0.2.15	74.125.133.189	TCP	54	51324 → 443 [ACK] Seq=3381 Ack=4463 Win=62780 Len=0
148	2.010606039	74.125.133.189	10.0.2.15	TLSv1.2	242	Application Data
149	2.010688707	10.0.2.15	74.125.133.189	TCP	54	51324 → 443 [ACK] Seq=3381 Ack=4651 Win=62780 Len=0
150	2.048947703	10.0.2.15	74.125.133.189	TLSv1.2	507	Application Data
151	2.049102787	10.0.2.15	74.125.133.189	TLSv1.2	227	Application Data
152	2.049336491	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=4651 Ack=3834 Win=65535 Len=0
153	2.049336570	74.125.133.189	10.0.2.15	TCP	60	443 → 51324 [ACK] Seq=4651 Ack=4007 Win=65535 Len=0
154	2.049406637	10.0.2.15	74.125.133.189	TLSv1.2	304	Application Data

Fig. 17. Wireshark results.

Figure 17 outlines a smooth connection between the virtual machine host (IP: 10.0.2.15) and the 'Google Cloud' server (IP: 74.125.133.189). Hence, it can be inferred that the Google servers sit in the middle between connected parties to prevent private network information from being passed between guests in a call. The same test was conducted on different days and from different host IP addresses with similar results, except the 'Google Cloud' server IP address would change. Knowing that 'Firefox' on 'Kali Linux' utilised TLSv1.2, an SSL review was conducted to ascertain TLS versions on older browsers in contrast to the most recent. The review used a hyperlink generated from Google Meet, which could be used to invite participants to a conference call. The results of the review showed that older browser versions utilising TLS 1.0, a protocol with several published vulnerabilities.

5.3 Browser Forensics for Google Meet

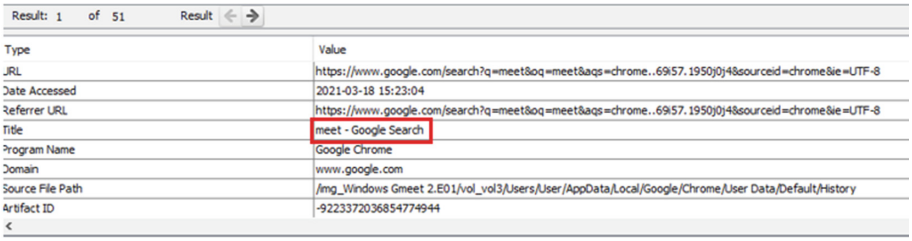
Browser forensics analyses the files stored locally on a system that correlate with the independent browsers. As a result, the disk image acquisition covers all files for each browser, permitting analyses in 'Autopsy'. An example of the 'Chrome' file storage system in 'Autopsy' can be shown in Fig. 18.

Cookies	2021-03-18 16:14:09 GMT	2021-03-18 16:14:09 GMT	2021-03-18 16:14:09 GMT	2021-03-18 15:22:34 GMT	32768
Cookies-journal	2021-03-18 16:14:09 GMT	2021-03-18 16:14:09 GMT	2021-03-18 16:14:09 GMT	2021-03-18 15:22:34 GMT	0
Favicons	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:22:27 GMT	49152
Favicons-journal	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:22:27 GMT	0
Google Profile Picture.png	2021-03-18 15:23:50 GMT	2021-03-18 15:23:50 GMT	2021-03-18 15:23:50 GMT	2021-03-18 15:23:50 GMT	1989
Google Profile.ico	2021-03-18 15:23:50 GMT	2021-03-18 15:23:50 GMT	2021-03-18 15:23:50 GMT	2021-03-18 15:22:29 GMT	181072
heavy_ad_intervention_opt_out.db	2021-03-18 15:23:21 GMT	2021-03-18 15:23:21 GMT	2021-03-18 15:23:21 GMT	2021-03-18 15:23:01 GMT	16384
heavy_ad_intervention_opt_out.db-journal	2021-03-18 15:23:21 GMT	2021-03-18 15:23:21 GMT	2021-03-18 15:23:21 GMT	2021-03-18 15:23:01 GMT	0
History	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:55:56 GMT	2021-03-18 15:22:26 GMT	135168
History Provider Cache	2021-03-18 15:23:28 GMT	2021-03-18 15:23:28 GMT	2021-03-18 15:23:28 GMT	2021-03-18 15:23:28 GMT	1450
History-journal	2021-03-18 15:56:28 GMT	2021-03-18 15:56:28 GMT	2021-03-18 15:56:28 GMT	2021-03-18 15:22:26 GMT	8720
Login Data	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:26 GMT	40960
Login Data For Account	2021-03-18 15:22:27 GMT	2021-03-18 15:29:03 GMT	2021-03-18 15:29:02 GMT	2021-03-18 15:22:26 GMT	40960
Login Data For Account-journal	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:26 GMT	0
Login Data-journal	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:27 GMT	2021-03-18 15:22:26 GMT	0
Media History	2021-03-18 15:34:36 GMT	2021-03-18 15:34:36 GMT	2021-03-18 15:34:36 GMT	2021-03-18 15:22:26 GMT	143360
Media History-journal	2021-03-18 15:34:36 GMT	2021-03-18 15:34:36 GMT	2021-03-18 15:34:36 GMT	2021-03-18 15:22:26 GMT	0

Fig. 18. Chrome file system in Autopsy.

The image file generated with the FTK Imager contained data for each of the tested browsers: 'Chrome', 'Firefox' and 'Edge'. When analysed with 'Autopsy', several artefacts were found. First, in the "History" SQLite database file, it is clear that "meet" was searched on Google, as shown in Fig. 19. After this has been searched, a result approximately ten minutes later in the "History" file shows that Google Meet was accessed (Fig. 20).

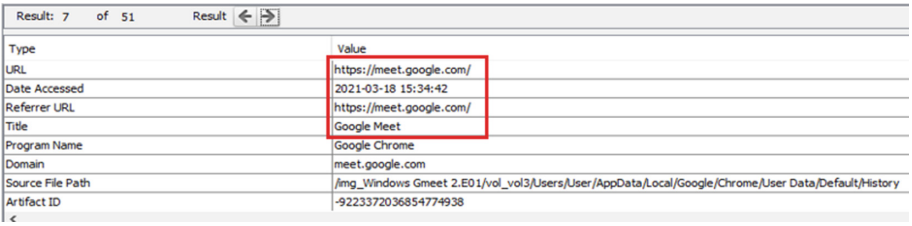
Furthermore, in the History file, a link that appears to be an invitation code can be found moments before Google Meet was accessed. It can be deduced that this is the invitational link used to access the specific meeting, as seen in Fig. 21. Also, remnants of information regarding the 'Google Calendar' can be found in the 'History' file. This tells us the calendar was accessed and contains information on potential future events that might occur within a specific week, as illustrated in Fig. 22. Within the "Web Data"



Result: 1 of 51 Result

Type	Value
URL	https://www.google.com/search?q=meet&oq=meet&aqs=chrome..69i57.1950j0j4&sourceid=chrome&ie=UTF-8
Date Accessed	2021-03-18 15:23:04
Referrer URL	https://www.google.com/search?q=meet&oq=meet&aqs=chrome..69i57.1950j0j4&sourceid=chrome&ie=UTF-8
Title	meet - Google Search
Program Name	Google Chrome
Domain	www.google.com
Source File Path	/img_Windows Gmeet 2.E01/vol_vol3/Users/User/AppData/Local/Google/Chrome/User Data/Default/History
Artifact ID	-9223372036854774944

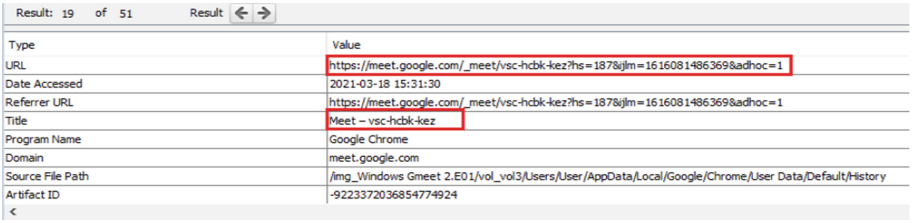
Fig. 19. Chrome – "meet" searched.



Result: 7 of 51 Result

Type	Value
URL	https://meet.google.com/
Date Accessed	2021-03-18 15:34:42
Referrer URL	https://meet.google.com/
Title	Google Meet
Program Name	Google Chrome
Domain	meet.google.com
Source File Path	/img_Windows Gmeet 2.E01/vol_vol3/Users/User/AppData/Local/Google/Chrome/User Data/Default/History
Artifact ID	-9223372036854774938

Fig. 20. Google Meet accessed.

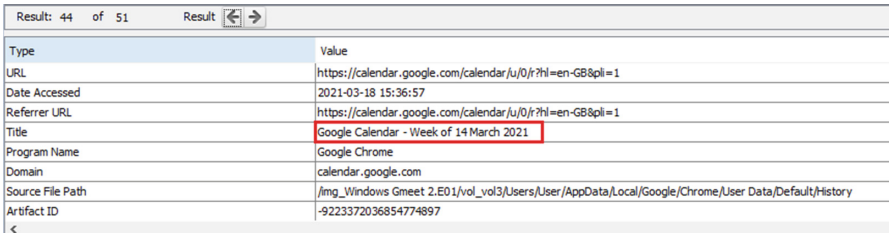


Result: 19 of 51 Result

Type	Value
URL	https://meet.google.com/_meet/vsc-hcbk-kez?hs=187&ijlm=1616081486369&adhoc=1
Date Accessed	2021-03-18 15:31:30
Referrer URL	https://meet.google.com/_meet/vsc-hcbk-kez?hs=187&ijlm=1616081486369&adhoc=1
Title	Meet – vsc-hcbk-kez
Program Name	Google Chrome
Domain	meet.google.com
Source File Path	/img_Windows Gmeet 2.E01/vol_vol3/Users/User/AppData/Local/Google/Chrome/User Data/Default/History
Artifact ID	-9223372036854774924

Fig. 21. Google Meet invitation code.

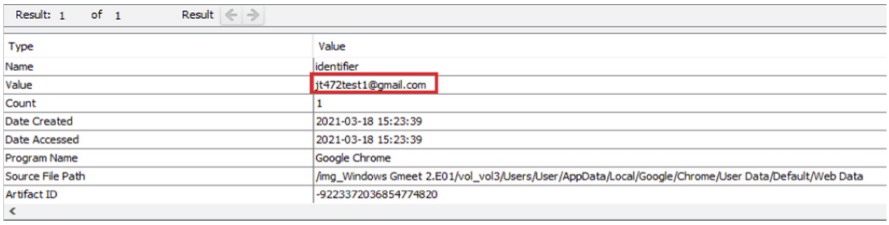
file in the browser files, an email address can be identified. This is the email address that was used to access the Google Meet call and can be seen being accessed shortly before using the 'Meet' application, this can be shown in Fig. 23.



Result: 44 of 51 Result

Type	Value
URL	https://calendar.google.com/calendar/u/0/r?hl=en-GB&pli=1
Date Accessed	2021-03-18 15:36:57
Referrer URL	https://calendar.google.com/calendar/u/0/r?hl=en-GB&pli=1
Title	Google Calendar - Week of 14 March 2021
Program Name	Google Chrome
Domain	calendar.google.com
Source File Path	/img_Windows Gmeet 2.E01/vol_vol3/Users/User/AppData/Local/Google/Chrome/User Data/Default/History
Artifact ID	-9223372036854774897

Fig. 22. Chrome – 'Google Calendar' artefact.



Result: 1 of 1 Result

Type	Value
Name	identifier
Value	jt472test1@gmail.com
Count	1
Date Created	2021-03-18 15:23:39
Date Accessed	2021-03-18 15:23:39
Program Name	Google Chrome
Source File Path	/img_Windows Gmeet 2.E01/vol_vol3/Users/User/AppData/Local/Google/Chrome/User Data/Default/Web Data
Artifact ID	-9223372036854774820

Fig. 23. Chrome – Email artefact.

The artefacts that have been obtained by 'Chrome' were also found in the browser files for both 'Firefox' and 'Edge'. In 'Firefox', the browsing history was located in an SQLite database labelled 'defaultplaces.sqlite' file and the email used to login to Google Meet was located in the 'defaultlogins.json' file. Similarly, for the Edge, browsing history was located in a file labelled 'History' and the email used to login to 'Meet' was located in 'Web Data'.

5.4 Evaluation of Findings in Google Meet

Two important findings emerged from the investigation of Google Meet. In the first finding, a collection of artefacts indicates that a user accessed and circumstantially used the Google Meet application. This information can be gathered from artefacts such as the "History" file, which shows when the web application was accessed. Additionally, this information can be backed up with data retrieved from the memory surrounding the networking activity, producing a time-labelled artefact that can be correlated with the "History" file data. Furthermore, the 'Web data' file contains the login email address, potentially identifying the user active at the time Google Meet was accessed.

The second finding relates to the hyperlinks recovered from the "History" file. With the hyperlink being unique to a specific call, there is a possibility of proving a person was involved in the Google Meet call without further proof required, as the retrieved link alone would be sufficient to show that the machine was used to access the specific call. Additionally, the hyperlinks obtained can be used to rejoin existing calls. Therefore, if a malicious person gained access to this artefact in the browser files, they could potentially gain access to a Google Meet call that they were not supposed to be on.

6 Conclusions

The Covid-19 pandemic introduced some difficult times for businesses and education to remain connected. With technology such as videoconferencing assisting in replicating some form of connection, this study examines the security of two popular applications for this purpose: Google Meet and MS Teams. After working through the stages of this study chronologically, beginning with the extensive research phase to identify gaps in the literature, this study can conclude that whilst Google Meet and MS Teams may be more cybersecure than similar applications in studies, it still presents a number of important cyber forensic artefacts that can be used to aid investigations or perhaps in a malicious

manner. Results do reveal several key artefacts, including suspects' email addresses, as well as email addresses of other parties who may have been involved. Finding out that these artefacts exist and knowing where to look for them could be key information for investigators, since it would save them time and resources.

Considering the scope of the study focused on only the 'Windows 10' operating system, the evidence may be limited. Therefore, it could be suggested that the use of different operating systems may present more, less or simply different artefacts. Future work should include testing the application on other popular platforms, such as, but not limited to, 'macOS', 'iOS', 'Linux' and 'Android'. By applying this study to alternative platforms, a full picture of the forensic soundness of both Google Meet and MS Teams can be created.

References

1. Statista: Growth in downloads of select video conferencing apps as of March 2020 vs. weekly average for Q4 2019, by country (2021). https://www.statista.com/statistics/1109875/download-growth-video-conferencing-apps/. Accessed 02 September 2021
2. Office for National Statistics: Coronavirus and homeschooling in Great Britain: April to June 2020 (2020). https://www.ons.gov.uk/peoplepopulationandcommunity/educationandchildcare/articles/coronavirusandhomeschoolingingreatbritain/apriltojune2020. Accessed 02 September 2021
3. Wiltshire Police: Incidents of 'zoom-bombing' reported in Wiltshire - Wiltshire Police (2020). https://www.wiltshire.police.uk/article/6136/Incidents-of-zoom-bombing-reported-in-Wiltshire. Accessed 02 September 2021
4. BBC News: 'Zoombombing' targeted with new version of app' (2020). https://www.bbc.com/news/business-52392084. Accessed 02 September 2021
5. Sky News: Coronavirus: FBI investigating after pornography used to 'Zoombomb' video conferences (2020). https://news.sky.com/story/coronavirus-fbi-investigating-after-pornography-used-to-zoombomb-video-conferences-11966712. Accessed 02 September 2021
6. Hawthorn, N.: McAfee Blogs -Top 10 Microsoft Teams Security Threats (2020). https://www.mcafee.com/blogs/enterprise/cloud-security/microsoft-teams-top-ten-security-threats/. Accessed 02 September 2021
7. John, A.S.: It's Not Just Zoom. Google Meet, Microsoft Teams, and Webex Have Privacy Issues, Too (2020). https://www.hawaii.edu/its/wp-content/uploads/sites/2/2020/05/Google-Meet-Microsoft-Teams-Webex-Privacy-Issues-Consumer-Reports.pdf. Accessed 02 September 2021
8. Mahr, A., Cichon, M., Mateo, S., Grajeda, C., Baggili, I.: Zooming into the pandemic! A forensic analysis of the Zoom Application. Foren. Sci. Int. Dig. Invest. **36**, 301107 (2021). https://doi.org/10.1016/j.fsidi.2021.301107
9. CVE - Request CVE IDs (2021). https://cve.mitre.org/cve/request_id.html#cna. Accessed 02 September 2021
10. CVE-2017–6517: Vulnerability Details: CVE-2017–6517 (2019). https://www.cvedetails.com/cve/CVE-2017-6517/. Accessed 02 September 2021
11. Idowu, S., Dominic, D., Okolie, S.O., Goga, N.: Security vulnerabilities of skype application artifacts: a digital forensic approach. Int. J. Appl. Inf. Syst. **12** (2019). https://doi.org/10.5120/ijais2019451784
12. StatCounter GlobalStats: Mobile Operating System Market Share Worldwide (2021). https://gs.statcounter.com/os-market-share/mobile/worldwide. Accessed 02 September 2021