UNIVERSITY OF CENTRAL OKLAHOMA

JACKSON COLLEGE OF GRADUATE STUDIES

AN ANALYSIS INTO THE EXPLOITATION OF THE POST-ATTENDEE URL FEATURE IN ZOOM WEBINAR REGARDING MALWARE TRANSMISSION

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

In partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE IN FORENSIC SCIENCE – DIGITAL FORENSICS

By

AUSTIN TREVOR CAULEY
Edmond, Oklahoma
2022

AN ANALYSIS INTO THE EXPLOITATION OF THE POST-ATTENDEE URL FEATURE IN ZOOM WEBINAR REGARDING MALWARE TRANSMISSION

By: AUSTIN T. CAULEY

A THESIS

APPROVED FOR THE W. ROGER WEBB FORENSIC SCIENCE INSTITUTE

Dr. Mark McCoy, Chair

Ms. Rachael Elliot

Myungah Park

Dr. Myung-Ah "Grace" Park

Acknowledgements

I would first like to thank my graduate advisor and chair of my thesis committee, Dr. Mark McCoy of the University of Central Oklahoma's Forensic Science Institute, for the countless hours that he dedicated to answering my countless inquiries and providing valuable insight that helped me move forward with this project. I would also like to thank the members of my thesis committee, Dr. Myung-Ah Park and Rachael Elliott, MS, for their patience with me and my numerous emails in addition to their unwavering support.

I am grateful to the entire faculty of the University of Central Oklahoma's Forensic Institute for their support and understanding of my work schedule, allowing me to work full-time while maintaining a full-time course load. In particular, I would like to thank Dr. James Creecy and Caitlin Porterfield for their sound advice when I needed it the most.

I also would like to thank my friends and classmates who not only supported me throughout this journey, but also helped give me the confidence to reach for the stars. I would not be where I am today were it not for their generosity and compassion. Additionally, I would like to thank Elizabeth Petts, my digital forensics partner in crime, for inviting me to the Zoom webinar that prompted this thesis in the first place.

Finally, I would like to thank my family. To my parents, Paul Cauley, Steve Clemens, and Linda Cauley, I want to thank you from the bottom of my heart for everything that you have done for me throughout my life and seeing the potential within me, even when I could not see it. To my brother, Tyler Cauley, I know I don't say this to you enough, but I am proud of the person you turned out to be, and I cannot wait to see what the future has in store for you.

Abstract

In response to the COVID-19 pandemic, large businesses and organizations relied on video conferencing applications such as Zoom to maintain public health guidelines due in part to their robust set of features to facilitate productive group events while maintaining social distancing recommendations. While Zoom has many features that can be found in similar video conferencing applications, Zoom also contains a plethora of unique and cutting-edge features to entice modern users. However, when new features are introduced, an inherent risk of vulnerability exploitation has the potential to overshadow the benefits of the feature. One such vulnerable feature within Zoom webinar that is often overlooked is the post-attendee URL, a feature that allows Zoom webinar hosts to set a URL that participants will be redirected to after joining. This study aims to showcase the vulnerabilities of this feature by utilizing URLs of malicious websites and direct download links of files to transmit malware to Zoom webinar participants of the desktop application version of Zoom webinar. This study will also provide an analysis of the residual digital artifacts that are left behind when this feature is utilized to provide digital forensic examiners with the ability to create a comprehensive timeline of events for cases involving this type of attack.

Table of Contents

List of Tables

List of Figures

List of Appendices

Chapter 1

Introduction

At the beginning of the COVID-19 global pandemic in 2020, large organizations such as corporations and universities searched for online applications that would allow workers and students respectively to fulfill their daily responsibilities in a virtual environment due to health and safety concerns. While video conferencing software is not a new piece of technology, many of the already existing applications were either locked behind a paid subscription or unable to support the required number of simultaneous users for a single session. It was not long, however, when the video conferencing application known as Zoom began to skyrocket in popularity. With a basic license that was completely free and a plethora of new features to entice potential users, Zoom quickly found its way into the attention of the masses. That is not to say, however, that Zoom has not had its fair share of controversies. One such example of a widely prevalent trend in Zoom's early days was "Zoom bombing," a term used to describe the act of a user outside of an organization joins their Zoom call due to a misconfiguration of security settings (Setera, 2020). "Zoom bombing" eventually reached a boiling point when the Federal Bureau of Investigations issued a press release regarding the use of photos of sexually exploited children that were used to disrupt over 195 reported Zoom meetings (2020). As a result, the digital forensics community has taken a keen interest in Zoom. Despite the interest generated in Zoom, many digital forensic examiners in the field have little information to turn to. With Zoom's popularity being less than two years old at the time of this writing, researchers simply have not had enough time to conduct a detailed analysis of the various features of Zoom’s desktop application even under the best of circumstances.

While Zoom contains a plethora of features that could potentially be of evidentiary value from a digital forensics perspective, such as the text chat and file transfer features, there is one feature in particular that, despite not being as popular as the former features, has the potential to unleash devastating consequences towards Zoom’s users: the post-attendee URL. According to Zoom, the post attendee URL is designed to “redirect participants to [an] organization’s website after [the participants] leave a Zoom meeting or webinar” (*Post-attendee URL*, p. 1). While this feature sounds beneficial at first, there are two major issues that are indicators that this feature can be exploited in unintended ways to create a negative experience to Zoom’s users. First, the documentation that Zoom Help Center provides for this feature does not have a single mention of basic security features that would typically be commonplace for applications with similar capabilities of redirecting users to a predetermined URL that is setup by a host. While the article does mention that organizations have to apply for a vanity URL for Zoom meetings, Zoom webinar currently does not have this restriction (*Post-attendee URL*, p. 2). As a result, the host of a Zoom webinar can, based on the verbiage of the article posted by the Zoom Help Center, set the post attendee URL to any kind of URL. This is especially alarming when bringing up the second issue regarding how the behavior of this feature could be exploited to make phishing URL links appear more authentic thanks in part to Zoom’s ability to directly email participants who are invited to or register for a Zoom webinar. *Figure 1* illustrates how a phishing attack typically occurs, while *Figure 2* visually represents how the post-attendee URL could be used for malicious use according to the Zoom Help Center article. Evidently, there are several similarities between *Figure 1* and *Figure 2*, most notably how the result can potentially be the same in both scenarios. Therefore, it is imperative that the current body of existing literature for digital forensic examiners should include a thorough analysis of this feature by examining how it can be exploited and the digital artifacts that are left behind.

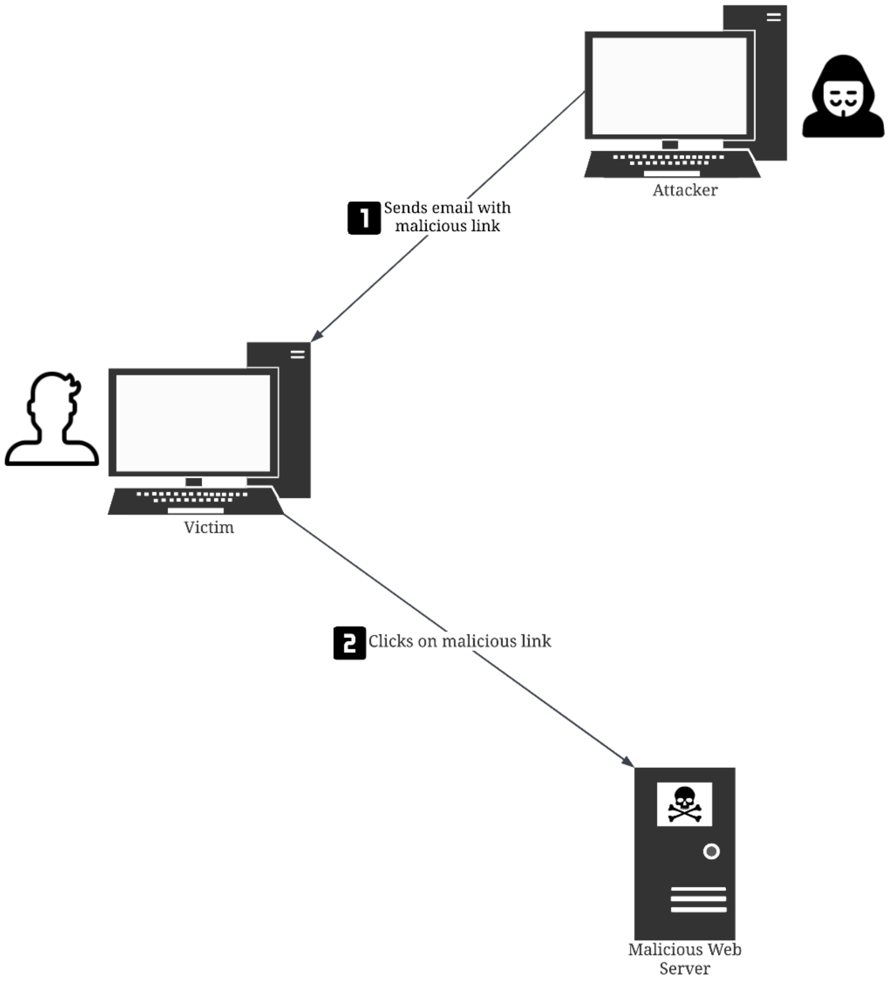


Figure 1: Visual Representation of a Phishing Attack

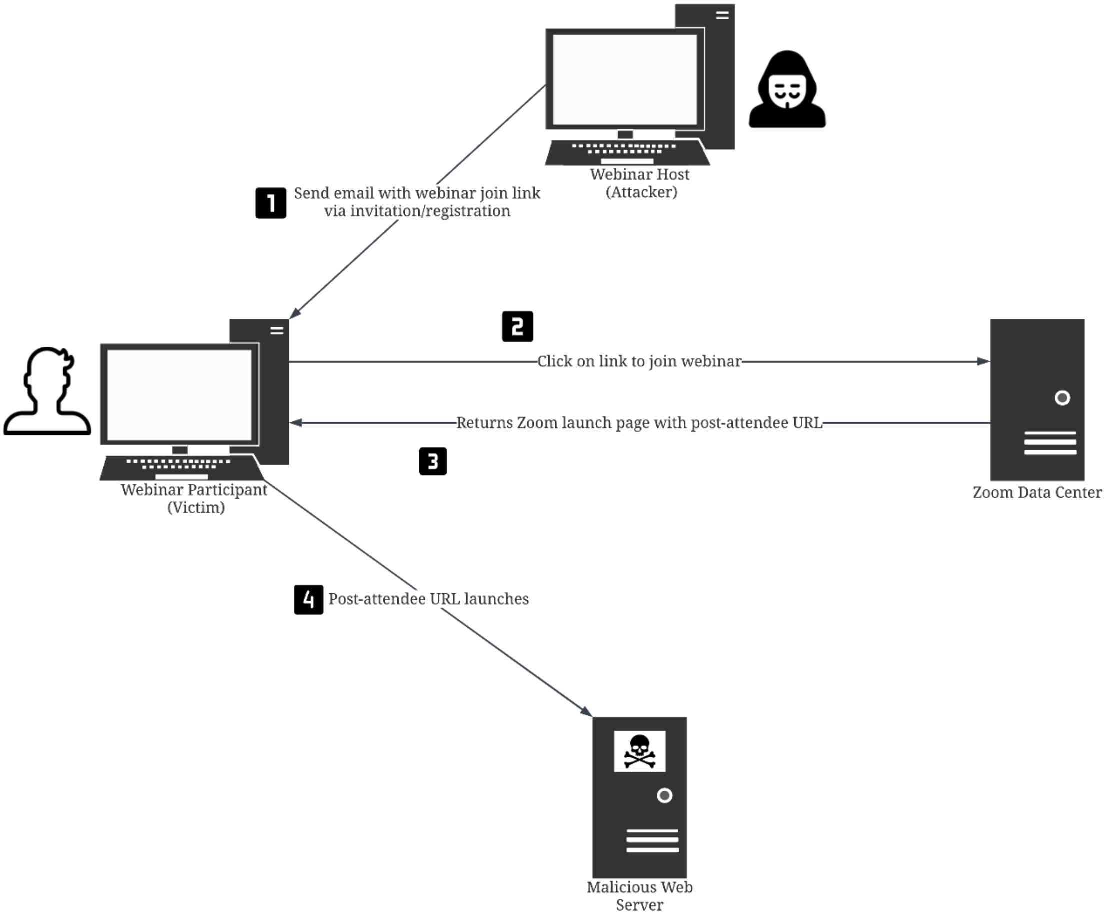

Webinar Host
(Attacker)
1 Send email with webinar join link via invitation/registration
2 Click on link to join webinar
Returns Zoom launch page with post-attendee URL 3
Webinar Participant
(Victim)
Zoom Data Center
4 Post-attendee URL launches
Malicious Web Server

Figure 2: Visual Representation of the Documented Post-Attendee URL Feature

Problem Statement

At the time of this writing, only one paper has been published regarding the analysis of digital artifacts created by Zoom. On March 2021, the journal *Forensic Science International: Digital Investigation* released a paper titled “Zooming into the pandemic! A forensic analysis of the Zoom Application” written by Andrew Mahr et al. In this paper, Mahr et al. claim that their paper is the first paper that aimed to analyze the Zoom application on desktops and mobile devices (2021, p. 1 - 2). While this study analyzes various digital artifacts from Zoom, there are certain features and artifacts that were not covered that could potentially be of forensic value to a digital forensic examiner, such as the post-attendee URL. The goal of this study is to build off the fundamental concepts that were provided by Mahr et al. and provide a deeper analysis that focuses on the functionality of the post-attendee URL and the digital artifacts that are generated using this feature.

Research Questions

To provide a detailed analysis on the post-attendee URL, the following two research questions were answered in this study:

1. How can the post-attendee URL be exploited to transmit malware to Zoom webinar participants without the participant taking any direct actions after joining the webinar?
2. What digital artifacts are created during the exploitation of the post-attendee URL that indicates a user was redirected to a certain URL via the post-attendee URL feature.

Chapter 2

Literature Review

Related video conferencing applications should be examined in order to view how past researchers conducted their research on a similar topic. Additionally, an examination of research frameworks that past researchers have used when conducting research in the field of digital forensics should be considered. Finally, an analysis into the Mahr's paper will be conducted that focuses on Mahr's findings to establish a known baseline of knowledge and determine how this study will build upon it.

Related Video Conferencing Applications

Before Zoom became a household name, there was another video conferencing application that came to mind for most people prior to the global pandemic: Skype. Similar to Zoom, Skype contains features that assist in fostering virtual connections, such as VoIP and text-based messaging (Nicoletti & Bernaschi, 2019, p. 160; Azab et al., 2015, p. 19 - 27). However, the differences begin to emerge very quickly. While Skype's network architecture was designed as a peer-to-peer (P2P) model, Zoom utilizes a cloud-based server architecture (Azab et al., 2015, p. 19 – 20; *Zoom Connection Process*, 2020, p. 1 - 2). Since one of the major benefits that the cloud-based server architecture offers is the ease of scalability, Zoom can easily host large-scale groups of users in a single meeting or webinar instance.

A study conducted by Nicoletti and Bernaschi, contrary to Mahr's paper, also go into detail regarding the Windows Registry (2019, p. 162). Nicoletti and Bernaschi utilized the Windows 10 Registry Editor in order to find important registry keys, such as information about the device, information about the user's account, and information about the user's most recent sign-in events (2019, p. 162 - 163). Since Mahr et al. focused on Zoom across multiple devices, including mobile devices such as Android and iOS, it is reasonable that the Windows Registry was not analyzed as it would only be found on Windows desktops. Nevertheless, the information found in the registry can play a key role in a digital forensic examination and will be investigated further in this study as it pertains to the Zoom desktop application. For a complete list of registry keys that can be setup by Zoom via Active Directory Administrative Template Configuration, see *Appendix A* (*Group Policy Options for the Windows desktop client and Zoom Rooms*, p. 2 - 11).

Digital Artifact Analysis Framework

When conducting research, one of the most important questions that a researcher can ask themselves would be "is the conducted research valid?" According to Christensen et al. (2014), validity is defined as "the overall probability of reaching the correct conclusion, given a specific method and data. Methods that are considered 'valid' give us the correct conclusion more often than chance…" (p. 124). To make sure that conducted research is valid, a proper framework needs to be established. This is where the Framework for Reliable Experimental Design (FRED) comes into play. FRED is an experimental framework that is specifically designed around digital forensics research to provide accuracy regarding the interpretation of data (Horsman 2018, p. 295). FRED consists of six stages that researchers must navigate through: planning, implementing, evaluating, repeating, analyzing, and confirming, as illustrated in *Figure 3* (Horsman, 2018, p.298 - 299).

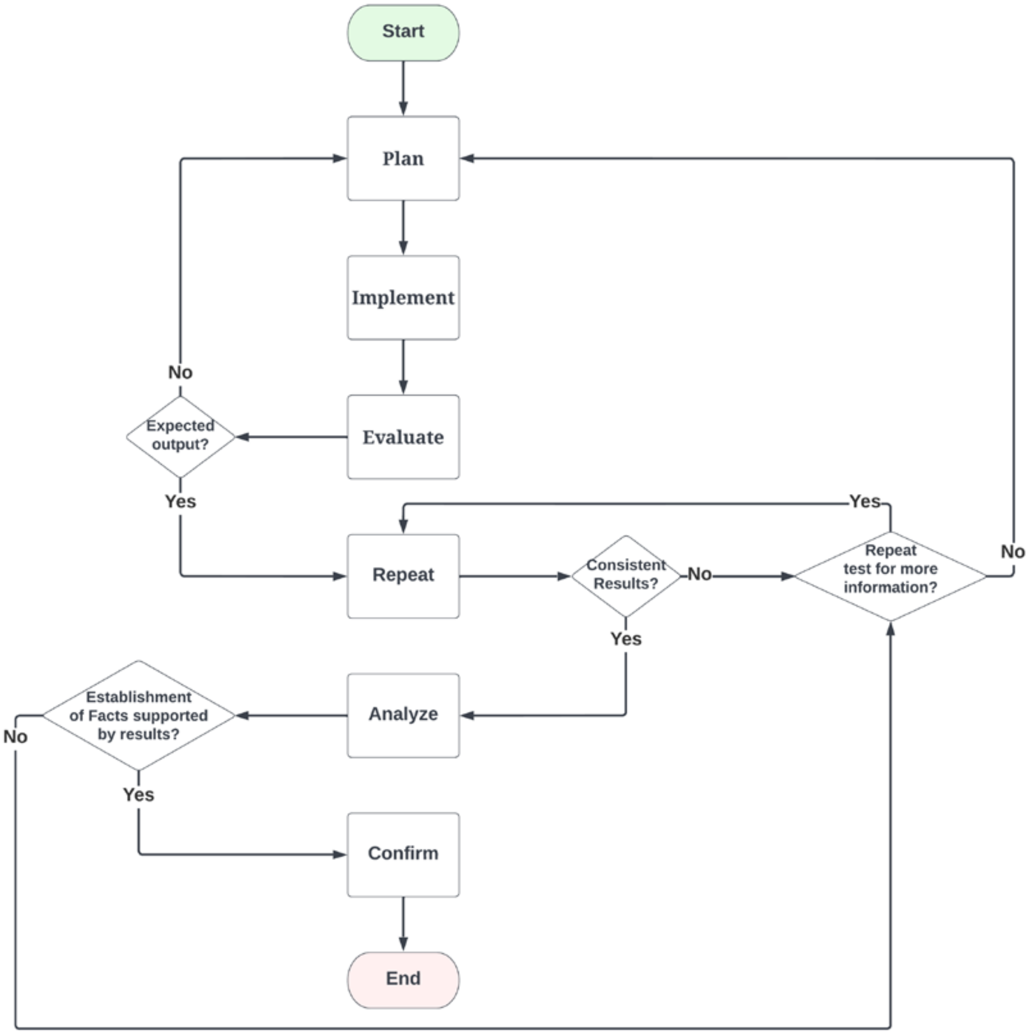


Figure 3: The FRED Research Framework

As shown in *Figure 3*, the FRED framework provides researchers with "checkpoints" to ask if the output is expected and consistent. If at this point that the answer to that question is 'no,' then there is a flaw in the experimental methodology that must be resolved. This framework is perhaps the most beneficial framework for this study, as it forces the researcher to continually keep the experimental design in mind and continually improve upon it if a flaw is discovered while evaluating the implementation.

Zoom Data Structure

As mentioned previously, this study aims to contribute to the paper submitted by Mahr et al. by providing a deeper analysis into features that were not analyzed regarding the windows desktop application. Table 1 provides a list of the artifacts-of-interest that Mahr et al. recorded for the windows desktop application for Zoom that are generated by certain features (2021, p. 4):

File Path	Account Type	Features
/AppData/Roaming/Zoom/data/“USER JID”@xmpp.zoom.us/”; USER JID”@xmpp.zoom.us.asyn.db	Basic and Licensed	Chats
/AppData/Roaming/Zoom/data/“USER JID”@xmpp.zoom.us/”; USER JID”@xmpp.zoom.us.db	Basic and Licensed	Contacts
/AppData/Roaming/Zoom/data/“USER JID”@xmpp.zoom.us/”; USER JID”@xmpp.zoom.us.idx.db	Basic and Licensed	Cached Data
/AppData/Roaming/Zoom/data/“USER JID”@xmpp.zoom.us/”; USER JID”@xmpp.zoom.us.sync.db	Basic and Licensed	Requests
/AppData/Roaming/Zoom/data/zoommeeting.db	Basic and Licensed	Encrypted Chats
/AppData/Roaming/Zoom/data/zoomus.db	Basic and Licensed	User Account Information
/AppData/Roaming/Zoom/data/“User JID”@xmpp.zoom.us/”; Hashed File Name”.db	Basic and Licensed	Temporary Webinar Database
/AppData/Roaming/Zoom/data/SSBAvatarCacheIndex.ini	Basic and Licensed	Avatar URL Cache Index
/AppData/Roaming/Zoom Plugin/ex2smtp.jsonLicensed	Licensed Only	Outlook Plugin JSON
/AppData/Roaming/Zoom Plugin/userSetting.json	Licensed Only	Outlook Plugin JSON
/AppData/Roaming/Zoom Plugin/alternateHosts.json	Licensed Only	Outlook Plugin JSON

Table 1: Data Path Directories and Files Found in the Windows Disk Drive

As evident in *Table 1*, this is not an exhaustive list of all the features and digital artifacts that Zoom, especially Zoom webinar, has to offer users. For example, the paper does not mention anywhere about the post-attendee URL, which may be used to navigate users to a website without any input from the user. Additionally, Zoom implemented a feature in version 5.6.1 that allows hosts and panelists to send files to attendees via chat. Since this paper came out prior to the implementation of this feature, this feature remains to be documented for future digital forensic examiners.

Summary

After reviewing the available literature for the Zoom desktop application for Windows 10, it is evident that the paper proposed by Mahr et al. provides a fundamental understanding of the Zoom data structure across multiple devices. However, a deeper analysis can be conducted to build off this understanding to gain a deeper understanding in the storage of user-generated data by Zoom. This study will accomplish this goal by focusing in primarily on the Windows 10 desktop application for Zoom and analyzing features that were not mentioned by Mahr et al., such as the post-attendee URL feature and file transfer via chat feature.

Chapter 3

Methodology

This study aimed to analyze the digital artifacts from the Zoom desktop application version 5.7.5 for the Windows 10 operating system version 2004. The main concern that was considered throughout this experiment was version control for the Zoom desktop application. Prior studies discovered that Zoom could force an update on the windows desktop application even if the user declines updates (Mahr et al., 2021, p. 2). With these key considerations kept in mind, this methodology is broken down into three distinct phases: environmental setup, data acquisition, and data analysis.

Environmental Setup

Before starting this experiment, 24 MyPassport 1TB hard drives were forensically wiped using a Tableau forensic duplicator in the University of Central Oklahoma Digital Forensics Laboratory for the purpose of creating sterilized forensic media. To ensure that all data was completely erased from the drives, the Tableau forensic duplicator ran a verification check to ensure that every byte on the hard drive was set to 0x00. After the drives were wiped and the verification was confirmed, 4 Dell OptiPlex desktops from the University of Central Oklahoma Digital Forensics Classroom were selected at random. These Dell OptiPlex desktops were disassembled and had their hard drive imaged using the Tableau forensic duplicators and four of the freshly wiped MyPassport 1TB hard drives. A verification check using the Tableau forensic duplicator's ability to generate and compare MD5 and SHA1 hash values was utilized to verify that the forensic image was created successfully.

After the four hard drives from the Dell OptiPlex desktops were successfully imaged, the desktops were reassembled to download the Zoom desktop application version 5.7.5. Once Zoom successfully installed and underwent the proper setup procedure, the Dell OptiPlex desktops were once again disassembled, imaged, and verified using the process mentioned in the previous paragraph. These forensic images accounted for any alterations made to the Dell OptiPlex desktops during the installation stage. After the Dell OptiPlex desktops were reassembled again, the desktops were ready to participate in Zoom webinars.

To account for the four types of users that can participate in a Zoom webinar, four user accounts were created and served as “participants” throughout this study, as indicated in *Table 2*. It is important to note that the role "co-host" is not a preassigned role compared to the other three listed roles. For a user account to be a co-host, they must first be invited as a panelist and then made a co-host upon entering the webinar (*Roles in a webinar*, p. 2 – 4). The host user account created a Zoom webinar with a post-attendee URL and invited the other three user accounts via email invitation. The remaining settings that were not related to the post-attendee URL were left in their default state. Once the other three user accounts accepted the invitation, the host officially started the webinar. After joining the webinar, all participant accounts waited for five minutes and observed the default web browser, Microsoft Edge, for indicators that the post-attendee URL launched successfully.

Account Name	**Account Type**	**User Account Role**
mmccoy@uco.edu	Education License	Host
cohostzoomtest@gmail.com	Basic/Free	Co-Host*
pannelistzoomtest@gmail.com	Basic/Free	Panelist
attendeezoomtest@gmail.com	Basic/Free	Attendee

Table 2: Zoom User Accounts and Roles

Once the host ended the webinar, the Dell OptiPlex desktops were disassembled and imaged using the Tableau forensic duplicators. Similar to the previous images, these images were verified using the MD5 and SHA1 hash algorithms that are supported by the Tableau forensic duplicator. For a visual representation of the experimental environment, please refer to *Figure 4*.

To provide a comprehensive analysis of the post-attendee URL, this environmental setup was replicated over the course of four trials. The first trial served to establish the baseline for what is considered normal behavior of the post-attendee URL when it is used properly. To achieve this, the post-attendee URL was set to www.uco.edu/fsi, a trusted and secure website, by the host with the remaining settings kept in their default state. The second and third trials were created to observe how the post-attendee URL function would behave when setting the post-attendee URL to a low security website that contained malware. For these trials, the post-attendee URL was set to www.bunnymeadow.com, a website that was created by the author of this study that contained a JavaScript file called *app.js* that contained a keylogger that transmitted keystrokes from the user and transmitted that data to a third part web server (See

Appendix B for the source code of the keylogger). Finally, the fourth trial consisted of determining the possibility of forcing a participant to directly download a file onto the desktop. Rather than using the URL of a website, the post attendee URL was set to a direct download link of https://drive.google.com/uc?export=download&id=11jIZzLh6fyURBNJcT6dzprZCrQ9dk9vS to determine if the post-attendee URL feature would execute the download sequence.

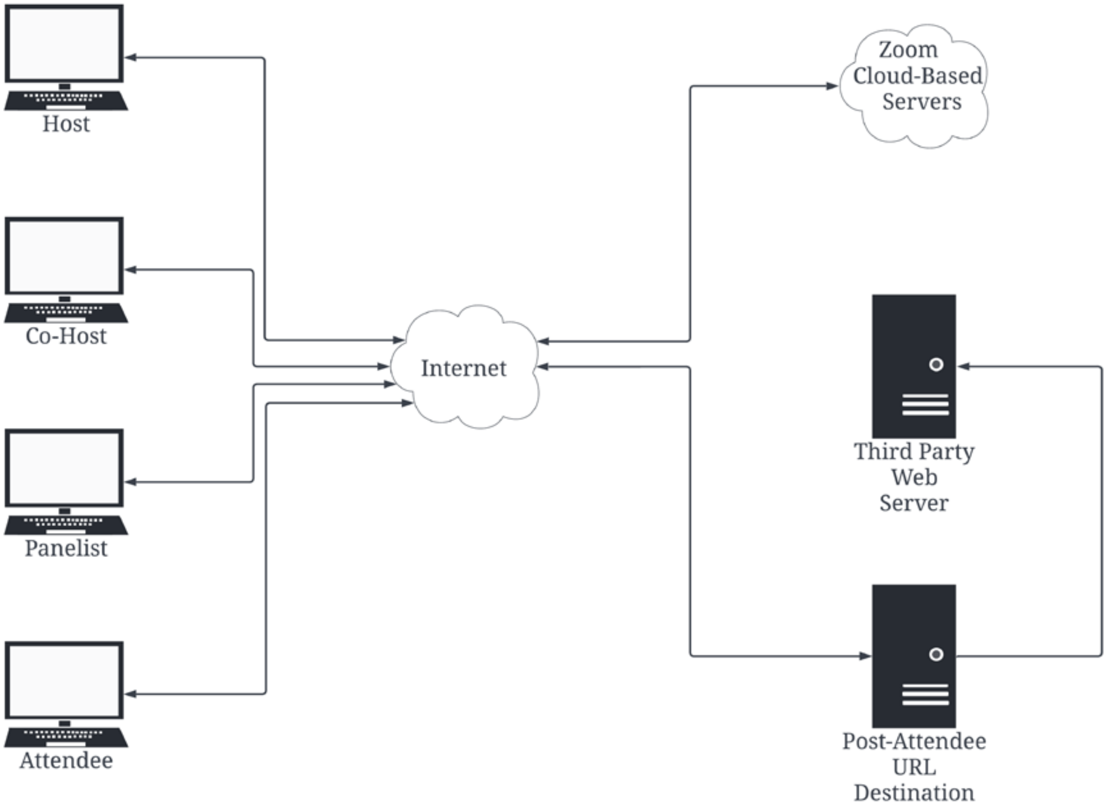


Figure 4: Visual Representation of the Experimental Environment

During these trials, the independent variable, which was the value that was set as the post-attendee URL, was manipulated to determine if Zoom webinar participants were redirected to malicious content solely due to changing the value of the post-attendee URL. As such, it was important that extraneous variables were identified to avoid altering the reported data in a meaningful way. During the planning stage of this study, the following extraneous variables were identified: the available user roles, the type of device each participant used, the operating system, the default browser, other webinar settings aside from the post-attendee URL, and the version of Zoom that was utilized. However, after conducting the second trial, an additional extraneous variable was identified by using the FRED research framework: the method in which participants joined the webinar. For a more detailed explanation as to how this extraneous variable was discovered, please refer to *Chapter 5: Discussion*.

Data Acquisition

For the purposes of capturing as much data as possible, live system acquisition techniques such as capturing the data from volatile memory such as the RAM were considered in addition to the forensic images of the Dell OptiPlex desktop hard drives that were created. To minimize the amount of data that was written on the Dell OptiPlex desktops, all external tools for the purpose of live system data acquisition were installed and executed on a 32 GB thumb drive. Additionally, all output files that were generated from these live acquisition tools were stored on the selfsame 32 GB thumb drive for the purpose of minimizing alterations to the Dell OptiPlex desktops.

As mentioned in the previous section, the Tableau forensic duplicators from the University of Central Oklahoma Digital Forensics Laboratory were used to create the forensic images throughout this experiment. However, once the Dell OptiPlex desktops were powered off, any data that was stored in the RAM would have been immediately erased. To prevent that data from disappearing, FTK Imager was used to create a memory capture file that contained the contents of the RAM at the time of the capture. Immediately after the webinar was ended by the host in each trial, FTK Imager was executed and captured the contents from the RAM of the Dell OptiPlex desktops into a .mem file, which were later opened by FTK and Magnet AXIOM for further analysis.

Data Analysis

As with any digital forensic investigation, the best analysis techniques involve a combination of manual analysis and the use of different tools. For the forensic images created with the Tableau forensic duplicator, FTK and Magnet AXIOM were used as an initial analysis step to generate a list of potential artifacts of interest. A manual examination was needed to verify which artifacts from the generated list contain forensically valuable information. The manual examination also served to discover artifacts that are difficult for FTK and Magnet AXIOM to parse, such as the case for undocumented keys in the Windows Registry. The .mem file that was captured from FTK Imager will also undergo the same treatment; an initial analysis using FTK and Magnet AXIOM, followed by a manual examination to confirm the results and search for artifacts that were not parsed properly by FTK and Magnet Axiom.

Chapter 4

Results

Throughout this experiment, four trials involving four unique webinars were created with default settings, sans the post-attendee URL. Each webinar was interacted with by four Windows 10 desktops that contained the Zoom desktop application. Each Windows 10 desktop had one of the four Zoom user accounts from *Table 2* signed in to the Zoom desktop application. In total, 24 forensic images were created. During each webinar, each webinar participant was scored on if the post-attendee URL executed after joining the webinar for five minutes, with each user role scored as “Yes” if the post-attendee URL did execute or “No” if the post-attendee URL did not execute.

Trial 1 Results

Trial 1 was conducted by setting the post-attendee URL as www.uco.edu/fsi. The host logged into the Zoom management portal through the web browser to start the meeting, which as a result caused the host to not join the webinar via email. All participants aside from the host were redirected to the post-attendee URL within five minutes of joining the webinar. See *Table 3* for a summary of the results for this trial.

User Role	**Zoom Join Method**	**Did Post-Attendee URL Execute**	**Browser**	**Zoom Version**
Host	Zoom Web Login	No	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Email Link	Yes	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

Table 3: Summary of Trial 1 Results

Trial 2 Results

Trial 2 was conducted by setting the post-attendee URL as www.bunnymeadow.com, a malicious website that the author of this study created which contained a JavaScript file that included a keylogger that transmitted keystrokes on the malicious webpage to a third-party server. For trial 2, the host logged in via email as the other participants. However, the panelist participant never received an email invitation to join the Webinar. As a result, the panelist joined by launching the Zoom application manually and entering the meeting ID and password into the app. The host, co-host, and attendee were redirected to the post-attendee URL, but the panelist was not. For the participants that were redirected to the post-attendee URL, a test was conducted to determine if the keylogger was functioning by asking the participants to type in a random word onto the webpage. The keylogger worked on all three participants who were redirected to the post attendee URL. See *Table 4* for a summary of the results for this trial.

User Role	Zoom Join Method	Did Post-Attendee URL Execute	Browser	Zoom Version
Host	Email Link	Yes	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Zoom Desktop Application	No	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

Table 4: Summary of Trial 2 Results

Trial 3 Results

Since the independent variable and an extraneous variable were altered in the second trial, trial 3 was conducted to replicate the results from trial 2 in order to confirm that the joining method of the panelist was the cause for the panelist participant to be exempt from the post-attendee URL redirection. The same URL was used as the post-attendee URL as was used in trial 2. Once all four participants joined the webinar via the invitation email link, all four participants were redirected to the post-attendee URL. To verify that the keylogger still functioned correctly for the participants, the same test from trial 2, only for this trial a different secret word was used. The keylogger successfully transmitted the data to the third-party server amongst all four webinar participants. See *Table 5* for a summary of the results from this trial.

In addition to the data collection procedure as outlined in *Chapter 3: Methodology*, the main JavaScript file, titled main.<hex string>.js, was collected from

Zoom's launch page that users interact with after clicking on the email invitation link in order to launch the Zoom desktop application. An excerpt of this source code for this JavaScript file as it pertains to the post-attendee URL can be found in *Appendix C*.

User Role	Zoom Join Method	Did Post-Attendee URL Execute	Browser	Zoom Version
Host	Email Link	Yes	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Email Link	Yes	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

Table 5: Summary of Trial 3 Results

Trial 4 Results

Trial 4 was conducted by setting the post-attendee URL to the value of https://drive.google.com/uc?export=download&id=11jIZzLh6fyURBNJcT6dzprZCrQ9dk9vS. Unlike the previous URLs that redirected to a specific website, this URL is a direct download link that, when executed, will begin downloading a photograph that is stored on the author's Google Drive. All four participants joined the webinar via the invitation email link and were redirected to the post-attendee URL five minutes after joining the webinar. However, instead of going to a specific website, the download process for the photograph attached to this direct download link started and was placed into the default download directory for each device. It's important to note that during this Trial, the Zoom desktop application forced an update that updated the client from version 5.7.5 to 5.9.3. Unfortunately, the client was unable to downgrade back to version 5.7.5. After reviewing the Windows release notes from Zoom's website, the authors determined that the post-attendee URL was not updated and remained unchanged. Nevertheless, precautions were taken and the JavaScript file from the Zoom launch page was collected again to compare it to the version that was collected in trial 3. Based on the comparison and the results from trial 4, it was determined that the post-attendee URL was unaffected by the update to version 5.9.3. See *Table 5* for a summary of the results from this trial. See *Table 6* for a summary of the results from this trial.

User Role	Zoom Join Method	Did Post-Attendee URL Execute	Browser	Zoom Version
Host	Email Link	Yes	Microsoft Edge	5.9.3
Co-Host	Email Link	Yes	Microsoft Edge	5.9.3
Panelist	Email Link	Yes	Microsoft Edge	5.9.3
Attendee	Email Link	Yes	Microsoft Edge	5.9.3

Table 6: Summary of Trial 4 Results

Forensic Images

Of the 24 forensic images that were created throughout the study, 14 forensic images were created from hard drives that had a successful post-attendee URL execution. Upon analysis, all 14 forensic images contained digital artifacts that indicated that users were redirected to the post-attendee URL located in the Microsoft Edge database file labeled “History.” This database file was located in the C:\Users\<USER_NAME>\AppData\Local\Microsoft\Edge\User Data\Default directory, where <USER_NAME> represents the username of the profile that was signed into the device at the time of the webinar. After using Magnet AXIOM for the initial analysis, the data inside the database file were exported into an Excel spreadsheet in order to conduct a manual examination.

After conducting a manual examination on the database file known as “History,” it was revealed that this database file contained not only a list of previously visited URLs that were used by Microsoft Edge, but also specified how the transition from one URL to another URL was executed by categorizing the transition in one of two ways: either as “typed,” meaning that a user manually typed the URL into the address bar of Microsoft Edge, or “linked,” which indicates that Microsoft Edge was redirected to that URL by another action other than a user manually typing out the URL in the address bar. If the transition was marked as “linked,” the database file will include the URL that Microsoft Edge was visiting prior to the transition and stores this data in a column labeled “from_visit” in the table labeled “visit.” It is important to note that the pertinent data is actually spread across two different tables, “urls” and “visits,” and needs to be cross-referenced in order to place the relevant data in a single table that is easier to interpret. For instance, the “visits” table contains the raw data for the columns “id” and “from_visit.” However, the data that is in the column “url” is an integer rather than a string of characters. This is because the number in that column actually corresponds to the column labeled “id” in the table “urls.”

Chapter 5

Discussion

For this study, the goal was to determine if the post-attendee URL feature within Zoom webinar could be exploited to transmit malware, and if so, what digital artifacts would be left behind. As demonstrated throughout trials 2, 3, and 4, there are currently little, if any, security protocols that are in-place to prevent Zoom webinar hosts to utilize the post-attendee URL feature in bad faith by sending webinar participants to compromised websites or forcing webinar participants to download malicious files. However, from the JavaScript file with over 12,000 lines of code that was recovered in the third trial (see *Appendix C* for the excerpt of the source code pertaining to the post-attendee URL launch instructions), there is a strong body of evidence that suggest that the post-attendee URL can only be executed when Zoom's launch page is opened after webinar participants join via the email invitation link. This is further corroborated with the results from the first two trials, as the only time the post-attendee URL failed to execute was when a webinar participant joined via a means other than the email invitation link. In other words, webinar participants who join a webinar manually from the Zoom desktop application by entering the meeting ID and passcode, when applicable, rather than clicking on the email invitation link directly will not be redirected to the post-attendee URL even if a post-attendee URL is set by a host.

The evidence also suggests that users can prevent the post-attendee URL from executing even if they joined by the email invitation link by closing the Zoom launch page after joining the webinar. Referring once again to *Appendix C*, it is clear that this

JavaScript file uses the JavaScript function *setTimeout()* to create a five minute time once the webinar participant joins the webinar. Once that five minute timer reaches zero, the variable *location.href*, which when assigned a value is treated as a command to navigate to the provided URL, is assigned the value of the post-attendee URL. This modus operandi is consistent with how the public documentation provided by Zoom about the post-attendee URL. However, by knowing that the five minute timer is called by the *setTimeout()* function, this means that the timer is not persistent whenever the state of the Zoom launch page is altered. In other words, if the Zoom launch page was closed, the process responsible for the *setTimeout()* process would be killed as well and the post-attendee URL will never execute, as long as the process was killed before the timer reached zero.

Regarding the digital forensic artifacts that are generated from the post-attendee URL, the "History" database file, as mentioned in the previous section, will provide evidence of the post-attendee URL occurring since a record will be created in that database file that indicates that the web browser transitioned from the Zoom launch page to the value of the post-attendee URL that was set by the host. Timestamps are also stored when a new record is created, which can be pivotal to a digital forensic investigator when creating a timeline of events.

Unfortunately, the Zoom database files that were mentioned in the 2021 paper published by Mahr et al. are now encrypted and the data contained within those files is unavailable to be examined without knowledge of the decryption keys. The memory capture was analyzed as an attempt to retrieve any of the keys in case they were stored in the RAM. Unfortunately, these keys were unable to be recovered.

Limitations

Throughout this study, several limitations arose that need to be addressed. Perhaps the most glaring limitation of this study was the choice to only use Microsoft Edge throughout this experiment. The reasoning behind this choice is twofold; first, Microsoft Edge comes preinstalled on modern Windows 10 desktops. By using Microsoft Edge, this allowed the baseline forensic images to be close to a fresh installation of a Windows 10 operating system, which minimized the number of software installations. Additionally, Microsoft Edge was chosen since it is a chromium-based browser. As a chromium-based browser, Microsoft Edge shares multiple characteristics with other chromium-based browsers, including how the “History” database file is stored. This would allow the result of the forensic examination to become pertinent to multiple web browsers. However, there is a case to be made that this study excludes non-chromium-based web browsers in the sense that the results may not be as relevant as it would be for chromium-based browsers.

Another evident limitation of this study was the use of only Windows 10 desktops without testing the post-attendee URL on other devices, such as Android and iOS devices, and other operating systems, such as Macs or Linux devices. This was mostly due to budgetary concerns, as the only type of experimental devices in the University of Central Oklahoma Digital Forensics Laboratory on-hand were Windows 10 desktops.

Future Work

With this study serving as a baseline in understanding how the post-attendee URL feature within Zoom webinar functions, future studies should be conducted that address the major limitations of this study. For example, future research could involve studying the behavior of the post-attendee URL feature on different devices, such as Android and iOS, and on different operating systems, such as Macs and Linux. While the results from this study can be used to postulate a potential outcome, there is currently not enough evidence at this time to state with certainty what the exact outcome would be.

In addition to studying the interaction between the post-attendee URL feature amongst different devices and operating system, it is also important to study the digital artifacts that are generated on non-chromium web browsers. While the concept of a browser history is commonplace amongst many modern web browsers, how that data is stored can be different from one browser to another. A comprehensive forensic analysis of an exhaustive list of non-chromium-based web browsers interacting with the post-attendee URL would be of great benefit to digital forensic examiners.

Finally, it is highly recommended that studying how Zoom encrypts and decrypts their database files would be of tantamount importance. While this study was unable to decrypt any of Zoom's database files, the ability to analyze the data stored within them could provide additional digital artifacts regarding the post-attendee URL. One potential starting point for researchers could be the file labeled "Zoom.us.ini" located in the same directory as the rest of Zoom's database files. Zoom.us.ini contains a field labeled "win_os_encrypt_key," but using the string as the key, even after decoding it as a base64 string, does not currently return any meaningful results.

Conclusion

Throughout the conducted trials, it is evident that the post-attendee URL contains multiple weaknesses that leave it vulnerable to exploitation by bad actors. From sending Zoom webinar participants to a malicious website to forcing a file to download on a participant's device, there is a startlingly lack of security protocols in place if a Zoom webinar host is acting in bad faith when creating the webinar and abusing the post-attendee URL feature. However, there are actions that webinar participants can take to mitigate this risk by avoiding the use of the email invitation link and instead manually launch the Zoom desktop application and enter the meeting ID and password. Joining a Zoom webinar with this method will avoid launching the Zoom launch page, which contains the JavaScript code that is responsible for redirecting participants to the post-attendee URL.

In addition to illustrating how the post-attendee URL feature can be exploited, this study also looked at the digital artifacts that were created to construct a timeline of events post-incident response. For this purpose, the "History" database file was analyzed due to its ability to maintain records that indicate how a user was directed to a particular URL. Additionally, JavaScript files from the Zoom launch page were analyzed to deconstruct how the post-attendee URL function behaves at a low level.

While this study successfully completed its objective of answering all three research questions that were proposed at the beginning of this thesis, there is still more information to the post-attendee URL that can be gleamed. For instance, this study did not investigate how the post-attendee URL functioned on other devices and operating systems outside of the Windows 10 desktops provided by the University of Central

Oklahoma Digital Forensics Laboratory, nor did it analyze the digital artifacts generated by the post-attendee URL feature on non-chromium-based browsers. While this study is important in understanding how the post-attendee URL feature functions and how it can be exploited, it is imperative that future researchers continue the pursuit of knowledge and continue to enhance our understanding of new and emerging technical trends for the sake of the digital forensics community.

References

Azab, A, Watters, P, & Layton, R. (2012). Characterizing Network Traffic for Skype Forensics. 2012 Third Cybercrime and Trustworthy Computing Worshop, 19–27. https://doi.org/10.1109/CTC.2012.14

Christensen, A. M., Crowder, C. M., Ousley, S. D., & Houck, M. M. (2013). Error and its Meaning in Forensic Science. *Journal of Forensic Sciences*, *59*(1), 123–126. https://doi.org/10.1111/1556-4029.12275

Federal Bureau of Investigations. (2020, May 20). *FBI Warns of Child Sexual Abuse Material Being Displayed During Zoom Meetings* [Press release]. https://www.fbi.gov/news/pressrel/press-releases/fbi-warns-of-child-sexual-abuse-material-being-displayed-during-zoom-meetings

Group Policy Options for the Windows desktop client and Zoom Rooms. (n.d.). Zoom Help Center. Retrieved April 19, 2021, from https://support.zoom.us/hc/en-us/articles/360039100051-Group-Policy-Options-for-the-Windows-Desktop-Client-and-Zoom-Rooms

Horsman, G. (2018). Framework for Reliable Experimental Design (FRED): A research framework to ensure the dependable interpretation of digital data for digital forensics. Computers & Security, 73, 294–306. https://doi.org/10.1016/j.cose.2017.11.009

Mahr, A, Cichon, M, Mateo, S, Grajeda, C, & Baggili, I. (2021). Zooming into the pandemic! A forensic analysis of the Zoom Application. Forensic Science International: Digital Investigation, 36. https://doi.org/10.1016/j.fsidi.2021.301107

Nicoletti, M, & Bernaschi, M. (2019). Forensic analysis of Microsoft Skype for Business. Digital Investigation, 29, 159–179. https://doi.org/10.1016/j.diin.2019.03.012

Post-attendee URL. (n.d.). Zoom Help Center. Retrieved April 20, 2021, from https://support.zoom.us/hc/en-us/articles/360000518526-Post-attendee-URL

Roles in a webinar. (n.d.). Zoom Help Center. Retrieved April 19, 2021, from https://support.zoom.us/hc/en-us/articles/360000252726-Roles-in-a-webinar

Setera, K. (2020, March 30). *FBI Warns of Teleconferencing and Online Classroom Hijacking During COVID-19 Pandemic*. Federal Bureau of Investigations. https://www.fbi.gov/contact-us/field-offices/boston/news/press-releases/fbi-warns-of-teleconferencing-and-online-classroom-hijacking-during-covid-19-pandemic

Zoom Connection Process. (2020, April). Zoom Video Communications Inc. Retrieved April 20, 2021, from https://explore.zoom.us/docs/doc/Zoom%20Connection%20Process%20Whitepaper.pdf

Appendix A: Registry Keys, Items, Values for Zoom

- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Zoom\Zoom Meetings\General

Registry Item	**Policy**	**Default Value**
AlwaysUsePersonalMeetingID	Always use personal meeting ID to start meeting for instant meetings	Disabled
AutoStartAfterReboot	Auto start client after reboot	Disabled
AutoStartInSystemTray	Auto start client after reboot in the system tray	Disabled
BlockUntrustedSSLCert	The client will block connections to untrusted SSL certificates	Disabled
CloseToSystemTray	The client will minimize to system tray and not show on taskbar when closed	Enabled
DisableCreatingDesktopShortcut	Disable creating a desktop shortcut	Disabled
DisableFacebookLogin	Disable login with Facebook OAuth	Disabled
DisableGoogleLogin	Disable login with Google OAuth	Disabled
DisableLoginWithSSO	Disable login with SSO	Disabled
DisableLoginWithEmail	Disable login with email and password	Disabled
DisableKeepSignedInWithSSO	Disable keep signed in if signing in with SSO	Disabled
DisableKeepSignedInWithGoogle	Disable keep signed in if signing in with Google	Disabled
DisableKeepSignedInWithFacebook	Disable keep signed in if signing in Facebook	Disabled

EnableClientAutoUpdate	Enable updates through the client by users. When disabled, the **Check for Updates** button is also hidden.	EXE: Enabled MSI: Disabled
ForceLoginWithSSO	Force login with SSO only	Disabled
SetAccountIDsRestrictedToJoin	Set account IDs that client is restricted to join a meeting hosted by specific Account ID numbers (separated by ",") Example: 11111,22222	No Value
SetEmailDomainsRestrictedToLogin	Restrict email domains that the client can log in with (separated by "&") Example: abc.com & zoom.us	No Value
SetSSOURL	Set default SSO URL for a client login Example: yourcompany.zoom.us	No Value
EnableEmbedBrowserForSSO	Uses embed browser in the Windows client for SSO	Disabled
ForceSSOURL	Set and lock the default SSO URL for a client login	No Value
KeepSignedIn	This will keep the user signed into the client when it is restarted	Disabled
SetWebDomain	Sets the web domain for logging in or joining a meeting, by default the values is https://zoom.us or https://zoom.com	No value

DefaultUsePortraitView	Default to Portrait Mode upon opening Zoom	Disabled
ProxyPAC	Set proxy server to client with PAC URL	No Value
ProxyServer	Set a proxy server for the client as named proxy. Example: server:port	No Value
ProxyBypass	Set proxy bypass rule for the client	No Value
EnforceSignInToJoin	Forces users to be signed in before joining a meeting from the app	No Value
EnablePhoneLogin	Enables logging in via phone authentication	No Value
EnableAlipayLogin	Enables logging in via Alipay authentication	No Value
EnableQQLogin	Enables logging in via QQ authentication	No Value
EnableWeChatLogin	Enables logging in via WeChat authentication	No Value
EmbedUserAgentString	Set to embed one specified user agent string for all HTTP requests from Zoom client application	No Value
EmbedDeviceTag	Set to embed one specified device tag string for all HTTP requests from Zoom client application	No Value
EnableAutoUploadDumps	Automatically send dump logs, when there is a critical-failure issue	Disabled
EnableTemporalDeNoise	Enables de-noising of video	Enabled
EnableGPUComputeUtilization	Allows the client to utilize GPU acceleration for video	Enabled

	processing	
EnableHardwareAccForVideoSend	Allows the client to utilize GPU acceleration for sending video	Enabled
EnableHardwareAccForVideoReceive	Allows the client to utilize GPU acceleration for receiving video	Enabled
ShareSessionDisableUDP	Forces Screen Sharing traffic over TCP instead of UDP.	Disabled
IntegrateZoomWithOutlook	Show Zoom contact status in Outlook, and sets Zoom as the default chat, meeting, and phone app in Outlook.	Disabled
SetAccEventsOptions	Sets what alerts will be read by a screen reader. The following options are available (enter the numeric value in the string): • IM chat received = 1 • Participant joined/left meeting = 2 • Participant joined/left waiting room = 4 • Audio muted/unmuted by host = 8 • Video stoped by host = 16 • Screen sharing started stoped by participant = 32 • Recording permission granted revoked = 64 • Pub meeting chat	No Value (all alerts will be read)

	received = 128 • Meeting chat received = 256 • In-meeting file uploaded = 512 • Closed captioning available = 1024 • Closed captioning available typing privilege granted revoked = 2048 • Host privilege granted/revoked = 4096 • Cohost privilege granted/revoked = 8192 • Remote control permission granted/revoked = 16384 • Livestream started/stopped = 32768 • Participant raised/lowered hand = 65536 • Q&A question received = 131072 • Q&A answer received = 262144 • Role changed to panelist = 524288 • Role changed to attendee = 1048576	
DisableDirectConnectionPK	Will disable the direct connection PK request by	Disabled

	the Zoom client	
DisableDirectConnection2Web	Will disable all direct connections to Zoom web service.	Disabled
EnableAutoUploadMemlogs	Zoom client will send critical-failure-logs to Zoom backend to improve the service.	Disabled
Disable3rdModuleVerify	Disables the check of the signature of the third-party library.	Disabled
SetDevicePolicyToken	Set device policy's token from web settings	No Value

- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Zoom\Zoom Meetings\Meetings

Registry Item	**Policy**	**Default Value**
AlwaysShowConnectedTime	The client will show meeting connected time	Disabled
AlwaysShowMeetingControls	The client will always show meeting controls	Disabled
AutoAdjustAudioSettings	The client will adjust audio settings automatically	Enabled
AutoEnableDualMonitor	Enable dual monitor if the system supports	Disabled
AutoHideNoVideoUsers	The client will hide users with video turned off in gallery View	Disabled
AutoJoinVoIP	Auto-connect audio with VoIP when joining a meeting	Disabled

ControlAllAppsWhenRemoteControl	All applications can be controlled during remote control session	Disabled
DisableAeroModeDuringShareScreen	Disable Aero mode when screen sharing on Windows 7 system	Enabled
DisableAnnotation	The client will disable and hide the ability to annotate over shared screen	Disabled
DisableCloudRecording	The client will disable and hide the cloud recording feature	Disabled
DisableLocalRecording	The client will disable and hide the local recording feature	Disabled
DisableMeetingChat	The client will disable and hide the in-meeting chat feature	Disabled
DisableMeetingFileTransfer	Client will disable and hide the in-meeting computer file-transfer feature.	Disabled
DisableMeeting3rdPartyFileStorage	Client will disable and hide the in-meeting 3rd party file transfer options.	Disabled
DisableReceiveVideo	The client will disable and hide the ability to receive video	Disabled
DisableShareScreen	The client will disable and hide the Share Screen feature	Disabled
DisableVideoCamera	The client will disable and hide the ability to send video	Disabled
EnableHDVideo	The client will capture and send camera video in HD 720p format	Disabled

EnterFullScreenWhenJoinMeeting	The client will automatically enter full-screen mode when joining a meeting	Disabled
EnterFullScreenWhenViewingSharedScreen	The client will enter full-screen mode when viewing the shared screen	Enabled
FitContentWhenViewingSharedScreen	The client will resize content to fit the window when viewing the shared screen	Enabled
MuteVoIPWhenJoinMeeting	The client will mute VoIP when joining a meeting	Disabled
PromptConfirmWhenLeaveMeeting	The client will prompt confirmation when leaving a meeting	Enabled
RecordingFilePath	Set path for local meeting recording files Example C:\Users\User\MyZoomRecordings	User\Document
TurnOffVideoCameraOnJoin	Turn off video camera when joining the meeting	Disabled
EnableMirrorEffect	The client will enable mirror effect of your video camera	Enabled
EnableHIDControl	The client will enable HID audio device control	Enabled
DisableComputerAudio	The client will disable computer audio	Disabled
EnableSplitScreen	The client will enable side-by-side mode	Disabled
ShowConfirmDialogWhenWebJoin	Client will notify an user when auto-joining a meeting from website	Disabled

DisableRemoteControl	During screen sharing, the person who is sharing is not able to allow others to control the shared content.	Disabled
DisableRemoteSupport	Disallow meeting host to provide 1:1 remote support to another participant. This option depends on breakout room settings.	Enabled
AlwaysShowVideoPreviewDialog	Always show video preview dialog when joining a video meeting	Disabled
DisableVirtualBkgnd	Disable virtual background for windows and mac	Disabled
EnableFaceBeauty	Enable touch up my appearance	Enabled
DisableWhiteBoard	Disable whiteboard feature	Disabled
EnableShareAudio	Enable share audio feature	Disabled
EnableShareVideo	Enable share video feature	Disabled
DisableDirectShare	Disable to share with Zoom Rooms	Disabled
DisableDesktopShare	Disable share desktop feature	Disabled
DisableAudioOverProxy	Disable audio media stream over a proxy server	Disabled
DisableVideoOverProxy	Disable video media stream over proxy server	Disabled
DisableSharingOverProxy	Disable screen sharing media stream over a proxy server	Disabled
SetUseSystemDefaultMicForVOIP	When enabled, Zoom will use the default microphone set in Windows	Disabled
SetUseSystemDefaultSpeakerForVOIP	When enabled, Zoom will use the default speaker set in Windows	Disabled
SetUseSystemCommunicationMicForVOIP	Set to use the system default communication microphone for VoIP	Disabled

SetUseSystemCommunicationSpeakerForVOIP	Set to use the system default communication speaker for VoIP	Disabled
BandwidthLimitUp	Allows the restriction of uplink traffic bandwidth from the Zoom Client	No value
BandwidthLimitDown	Allows the restriction of downlink traffic bandwidth from the Zoom Client	No value
EnableIndependentDataPort	If enabled, the client will use the following ports for media transmission: AUDIO: 8803 AS: 8802 VIDEO: 8801	Disabled
LegacyCaptureMode	Disables GPU acceleration	Disabled
EnableStartMeetingWithRoomSystem	Displays the "Call Room System" button on the home screen of the Zoom app	Disabled
ShowZoomWinWhenSharing	Displays the Zoom meeting window, even when screen sharing	Disabled
MuteWhenLockScreen	When the computer is locked, if the Zoom client is in a meeting, it will automatically mute the microphone and turn off camera	Disabled
EnableOriginalSound	Automatically enables the Orginal Audio setting for the microphone	Disabled
AppendCallerNameForRoomSystem	If enabled the Zoom app will display the caller name for Room system	Disabled
EnableElevateForAdvDSCP	If enabled, the Zoom app will start an elevated video process to support advanced DSCP	Disabled

	marks	
EnableSpotlightSelf	Enables spotlight of your video when speaking	Disabled
Enable49video	Allows Gallery View to display up to 49 participants per screen	Disabled
EnableRemindMeetingTime	Enables a reminder for upcoming meetings	Disabled
VideoRenderMethod	Sets the specified video rendering method using the following string variables: 0 - Auto 1 - Direct3D11 Flip Mode 2 - Direct3D11 3 - Direct3D9 4 - GDI	0
PresentToRoomOptimizeVideo	The client will optimize screen sharing for video clip when user shares directly to a Zoom Room.	Enabled
PresentToRoomOption	Set specified sharing option when a user shares directly to a Zoom Room: 0 - Show all sharing options 1 - Automatically share desktop	1
PresentInMeetingOption	Set specified sharing option when user shares screen in a meeting: 0 - Show all sharing options 1 - Automatically share	1

	desktop	
DisableVideoFilters	Set to disable virtual filters	Disabled
ScreenCaptureMode	Set the specified screen capture mode: 0 - Auto 1 - Legacy operating systems 2 - Share with window filtering 3 - Advanced share with window filtering 4 - Advanced share without window filtering	0
HidePhoneInComingCallWhileInMeeting	Enable to prevent incoming call notifications while in a meeting.	Disabled
EnableShareClipboardWhenRemoteControl	Enable to allow clipboard access during remote control sessions.	Disabled
EnableDoNotDisturbInSharing	Set to silence system notifications when sharing desktop	Enabled
SetSuppressBackgroundNoiseLevel	Set noise suppression level Auto - 0 Low -1 Medium - 2 High - 3	0 (Auto)

SetAudioSignalProcessType	Set audio signal processing for Windows audio devices Auto - 0 Off - 1 On - 2	0 (Auto)

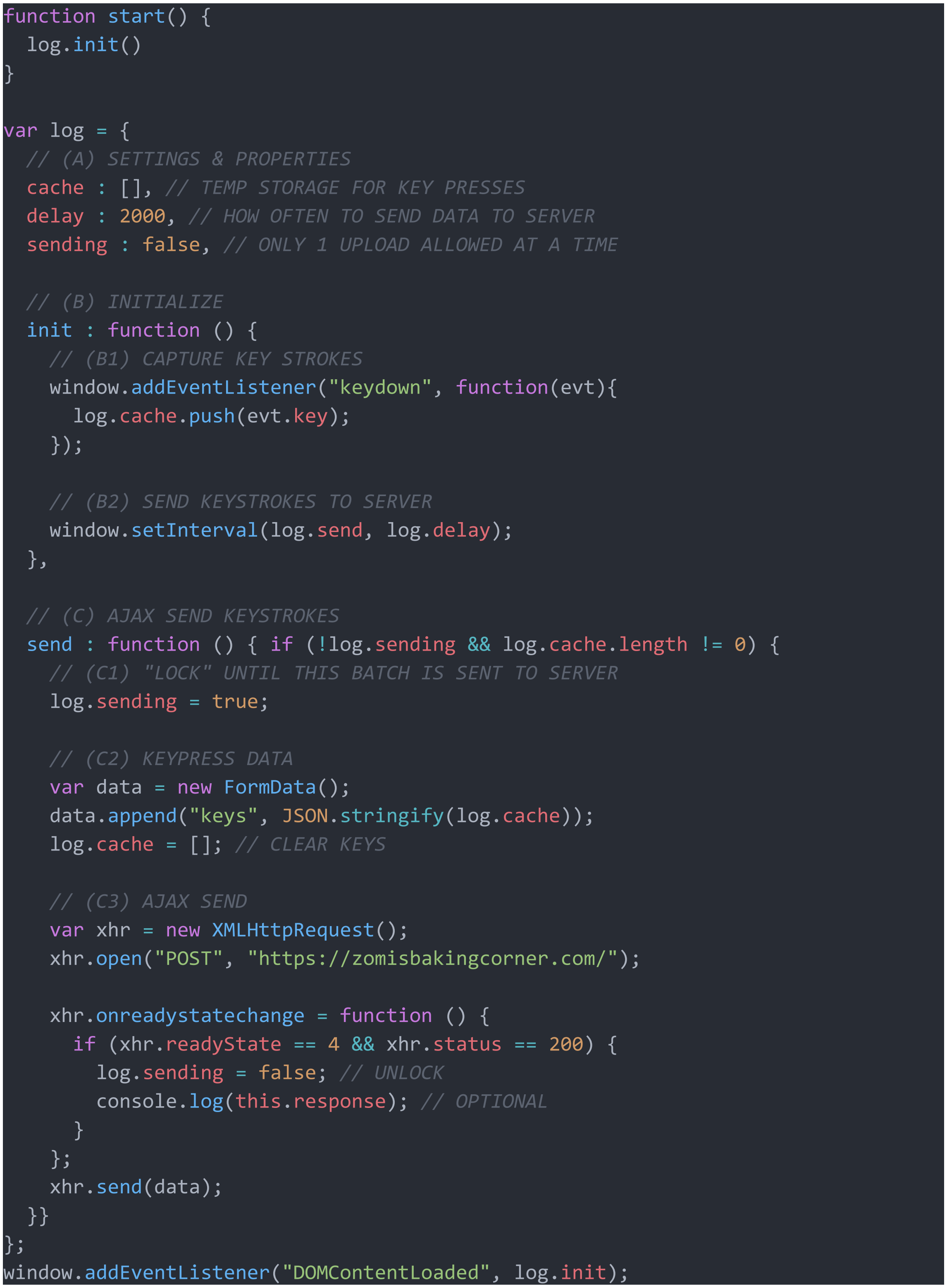
- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Zoom\Zoom Meetings\chat

Registry Item	**Policy**	**Default Value**
DisableLinkPreviewInChat	The client will disable and hide link preview feature in chat	Disabled
SetMessengerDoNotDropThread	Moves messages with new replies to the bottom of the chat window	Disabled
ShowVoiceMessageButton	Displays the Voice to Text option in chat controls	Enabled

- HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Zoom\Zoom Rooms\General

Registry Item	Policy	Default Value
EnableSSLVerification	The client will verify the server certificate	Enabled

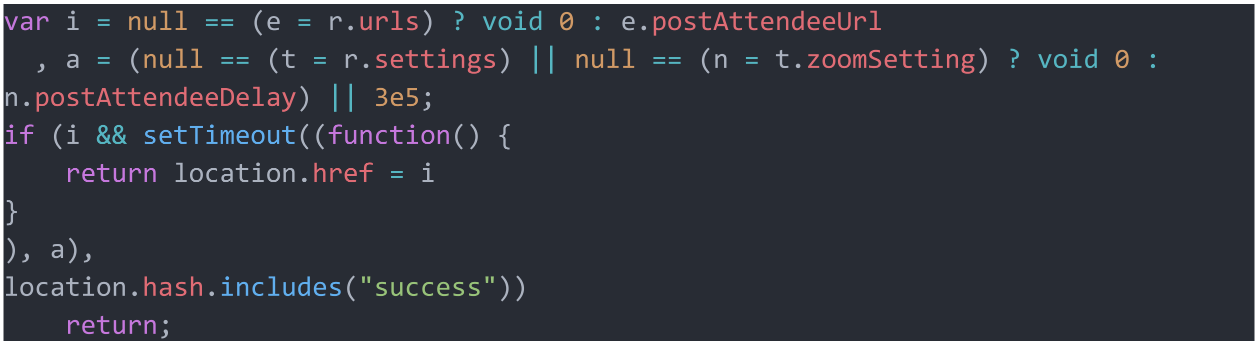
Appendix B: Source Code for *app.js*



```
function start() {
  log.init()
}

var log = {
  // (A) SETTINGS & PROPERTIES
  cache : [], // TEMP STORAGE FOR KEY PRESSES
  delay : 2000, // HOW OFTEN TO SEND DATA TO SERVER
  sending : false, // ONLY 1 UPLOAD ALLOWED AT A TIME

  // (B) INITIALIZE
  init : function () {
    // (B1) CAPTURE KEY STROKES
    window.addEventListener("keydown", function(evt){
      log.cache.push(evt.key);
    });

    // (B2) SEND KEYSTROKES TO SERVER
    window.setInterval(log.send, log.delay);
  },

  // (C) AJAX SEND KEYSTROKES
  send : function () { if (!log.sending && log.cache.length != 0) {
    // (C1) "LOCK" UNTIL THIS BATCH IS SENT TO SERVER
    log.sending = true;

    // (C2) KEYPRESS DATA
    var data = new FormData();
    data.append("keys", JSON.stringify(log.cache));
    log.cache = []; // CLEAR KEYS

    // (C3) AJAX SEND
    var xhr = new XMLHttpRequest();
    xhr.open("POST", "https://zomisbakingcorner.com/");

    xhr.onreadystatechange = function () {
      if (xhr.readyState == 4 && xhr.status == 200) {
        log.sending = false; // UNLOCK
        console.log(this.response); // OPTIONAL
      }
    };
    xhr.send(data);
  }}
};
window.addEventListener("DOMContentLoaded", log.init);
```

Appendix C: Excerpt From the Source Code of

meeting.<HEX_STRING>.js (Lines 6394 – 6401)



```
var i = null == (e = r.urls) ? void 0 : e.postAttendeeUrl
  , a = (null == (t = r.settings) || null == (n = t.zoomSetting) ? void 0 :
n.postAttendeeDelay) || 3e5;
if (i && setTimeout((function() {
    return location.href = i
}
), a),
location.hash.includes("success"))
    return;
```

Appendix D: Journal of Forensic Sciences Manuscript

Exploiting the Post-Attendee URL Feature in Zoom Webinar to Distribute Malware: Implications for Digital Forensics

Austin Cauley and Mark McCoy

Forensic Science Institute, University of Central Oklahoma, 100 N. University Dr, Edmond, 73034, Oklahoma, United States

Abstract

The post-attendee URL feature within the video conferencing application known as Zoom is often overlooked by digital forensic experts as a potential attack vector for malware transmission. However, with the ability to redirect webinar participants to any URL set by the host for the webinar, the post-attendee URL can be abused by bad actors to expose webinar participants to malicious websites or, in the worst-case scenario, force participants to download a file through the use of a direct download link URL. This study aims to showcase how this exploit can be replicated by creating an experimental environment involving four Windows 10 desktops running Zoom version 5.7.5 and creating a webinar with four user accounts acting as webinar participants and setting the post-attendee URL value to the URL of a website that contained a keylogger. In another trial, the same experimental environment was utilized, with the only difference beingthe post-attendee URL that was set to redirect webinar participants to a down- load link for a *.jpg* file. In both instances, every user account that joined the webinar via clicking on the invitation link that was emailed to each user account after registering for the webinar was redirected to the post- attendee URL regardless of their user account role. These results not only prove that the post-attendee URL can be exploited, but also provide insight as to how this type of attack can be prevented.

KEYWORDS

malware, Zoom, exploitation, vulnerabilities, phishing, digital artifacts

Highlights

- The post-attendee URL feature can be exploited to send Zoom webinar participants to malicious websites.
- The post-attendee URL feature can be exploited to force Zoom webinar participants to download a file via a direct download link
- By terminating the Zoom launch page after joining a Zoom webinar, the post-attendee URL can be prevented from executing.

Introduction

At the start of the 2020 COVID-19 global pandemic, corporations and universities alike searched for programs that would allow workers and students respectively to fulfill their daily responsibilities in a virtual environment due to health and safety concerns. It was not long, however, when the video conferencing application known as Zoom began to skyrocket in popularity. However, as the number of Zoom's users rapidly increased, several concerns about the security of this program were starting to propagate. One such example of a widely prevalent trend in Zoom's early days was "Zoom bombing," a term used to describe the act of a user outside of an organization joins their Zoom call due to misconfigured security settings [1]. "Zoom bombing" eventually reached a boiling point when the Federal Bureau of Investigations issued a press release regarding the use of photos of sexually exploited children that were used to disrupt over 195 reported Zoom meetings [2]. As a result, the digital forensics community has taken a keen interest in Zoom's capabilities for generating and storing digital artifacts.

One such feature within Zoom that has the potential to unleash devastating consequences towards Zoom's users, despite the feature's obscurity, is the post-attendee Uniform Resource Locator (URL). According to Zoom, the post attendee URL is designed to "redirect participants to [an] organization's website after [the participants] leave a Zoom meeting or webinar" [3]. While this feature sounds beneficial at first, there are two major issues that are indicators that this feature can be exploited in unintended ways to create a negative experience to Zoom's users. First, the documentation published by the developers of Zoom, Zoom Video Communications, Inc., does not mention any basic security features that would typically be commonplace for applications with similar capabilities of redirecting users to a predetermined URL. While the article does mention that organizations have to apply for a vanity URL for Zoom meetings, Zoom webinar currently does not have this restriction [3]. As a result, the host of a Zoom webinar can, based on the verbiage of the article posted by the Zoom Help Center, set the post attendee URL to any kind of URL. This is especially alarming when bringing up the second issue regarding how the behavior of this feature could be exploited to make phishing URL links appear more authentic thanks in part to Zoom's ability to directly email participants who are invited to or register for a Zoom webinar. Figure 1 illustrates how a phishing attack typically occurs, while Figure 2 visually represents how the post-attendee URL functions according to the Zoom Help Center article. Clearly, there are several similarities between Figure 1 and Figure 2, most notably how the result can potentially be the same in both scenarios. Therefore, it is imperative that the current body of existing literature for digital forensic examiners should include a thorough analysis of this feature by examining how it can be exploited and the digital artifacts that are left behind, as the current available literature is lacking in this regard.

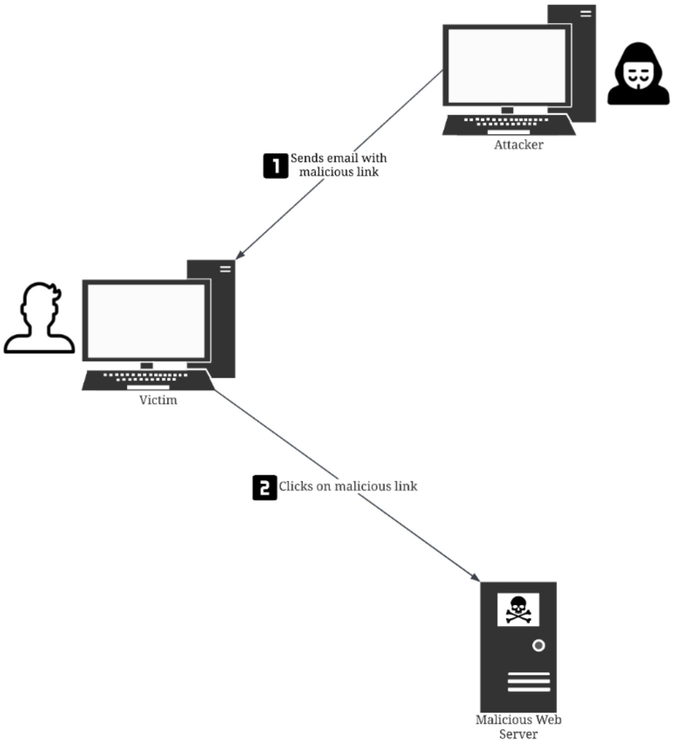


Figure 1: Visual Representation of a Phishing Attack

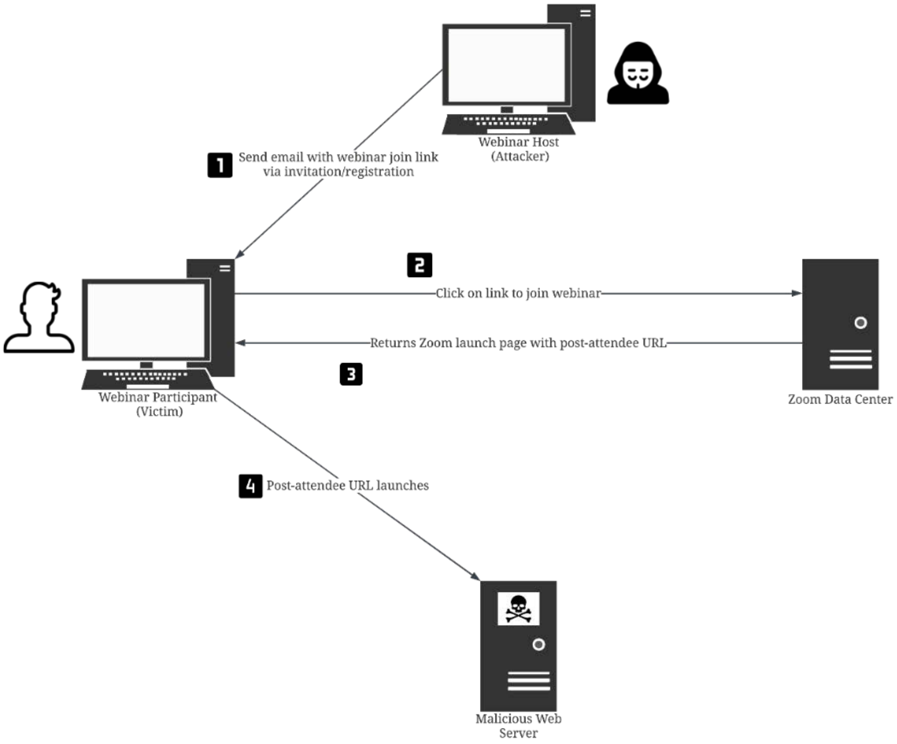

Webinar Host (Attacker)
1 Send email with webinar join link via invitation/registration
2
Click on link to join webinar
Returns Zoom launch page with post-attendee URL
3
Webinar Participant (Victim)
Zoom Data Center
4 Post-attendee URL launches
Malicious Web Server

Figure 2: Visual Representation of the Documented Post-Attendee URL Feature

Currently, only one paper has been published regarding the analysis of digital artifacts created by Zoom. On March 2021, the journal *Forensic Science International: Digital Investigation* released a paper titled "Zooming into the pandemic! A forensic analysis of the Zoom Application" written by Andrew Mahr et al. In this paper, Mahr et al. claim that their paper is the first paper that aimed to analyze the Zoom application on desktops and mobile devices [4]. While this study analyzes various digital artifacts from Zoom, there are certain features and artifacts that were not covered that could potentially be of forensic value to a digital forensic examiner, such as the post-attendee URL. The goal of this study is to build off the fundamental concepts that were provided by Mahr et al. and provide a deeper analysis that focuses on the functionality of the post-attendee URL.

Materials and Methods

For this experiment, four trials involving four unique webinars were created with default settings and adding the post-attendee URL. In each webinar, four desktops with a new installation of Windows 10 were used as the webinar participants. Prior to the start of the first trial, 24 1TB hard drives were forensically wiped for the purpose of creating sterilized forensic media. These sterilized hard drives served as the storage medium for the forensic images taken from each of the four desktops that joined the webinar in order to confirm that the desktop accessed the post-attendee URL via web-based digital artifacts such as the Microsoft Edge browser history. To ensure that all data was completely erased from the sterilized media, a checksum64 verification check was performed to ensure that every byte on the hard drive was set to 0x00. After the sterilized drives were successfully wiped and the verification was confirmed, each desktop was disassembled and had their hard drive imaged to four of the freshly wiped sterilized drives for the purpose of creating a baseline image of each of the desktops prior to installing the Zoom desktop application. MD5 and SHA1 hash values were generated from the original drive and the newly imaged drive and the results were compared to each other to verify that the forensic image was created successfully.

After the four hard drives from the desktops were successfully imaged, the desktops were reassembled to download the Zoom desktop application version 5.7.5. Once Zoom successfully installed and underwent the proper setup procedure, the desktops were once again disassembled, imaged, and verified using the process mentioned in the previous paragraph. These forensic images accounted for any alterations made to during the installation of the Zoom desktop application. After the desktops were reassembled again, the desktops were ready to participate in Zoom webinars.

To account for the four types of users that can participate in a Zoom webinar, four user accounts were created and served as “participants” throughout this study, as indicated in *Table 2*. It is important to note that the role "co-host" is not a preassigned role compared to the other three listed roles. In order for a user account to be a co-host, they must first be invited as a panelist and then made a co-host upon entering the webinar [8]. The host user account created a Zoom webinar with a post-attendee URL and invited the other three user accounts via email invitation. The remaining settings that were not related to the post-attendee URL were left in their default state. Once the other three user accounts accepted the invitation, the host officially started the webinar. After joining the webinar, all participant accounts waited for five minutes and observed the default web browser, Microsoft Edge, for indicators that the post-attendee URL launched successfully.

TABLE 1

Zoom User Accounts and Roles

Account Name	**Account Type**	**User Account Role**
hostzoomtest@gmail.com	Education License	Host
cohostzoomtest@gmail.com	Basic/Free License	Co-Host
pannelistzoomtest@gmail.com	Basic/Free License	Panelist
attendeezoomtest@gmail.com	Basic/Free License	Attendee

Once the host ended the webinar, the Dell OptiPlex desktops were disassembled and imaged using the Tableau forensic duplicators. Similar to the previous images, these images were verified using the MD5 and SHA1 hash algorithms that are supported by the Tableau forensic duplicator. For a visual representation of the experimental environment, refer to Figure 3.

To provide a comprehensive analysis of the post-attendee URL, this environmental setup was replicated over the course of four trials. The first trial served to establish the baseline for what is considered normal behavior of the post-attendee URL when it is used properly. To achieve this, the post-attendee URL was set to a trusted and secure website, by the host with the remaining settings kept in their default state. The second and third trials were created to observe how the post-attendee URL function would behave when setting the post-attendee URL to a low security website that contained malware. For these trials, the post-attendee URL was set to www.bunnymeadow.com, a website that was created by one of the authors of that contained a JavaScript file called *app.js* that contained a keylogger that transmitted keystrokes from the user andtransmitted that data to a third part web server. Finally, the fourth trial consisted of determining the possibility of forcing a participant to directly download a file onto the desktop. Rather than using the URL of a website, the post attendee URL was set to a direct download link of a picture that was hosted on Google Drive to determine if the post-attendee URL feature would execute the download sequence of the image file.

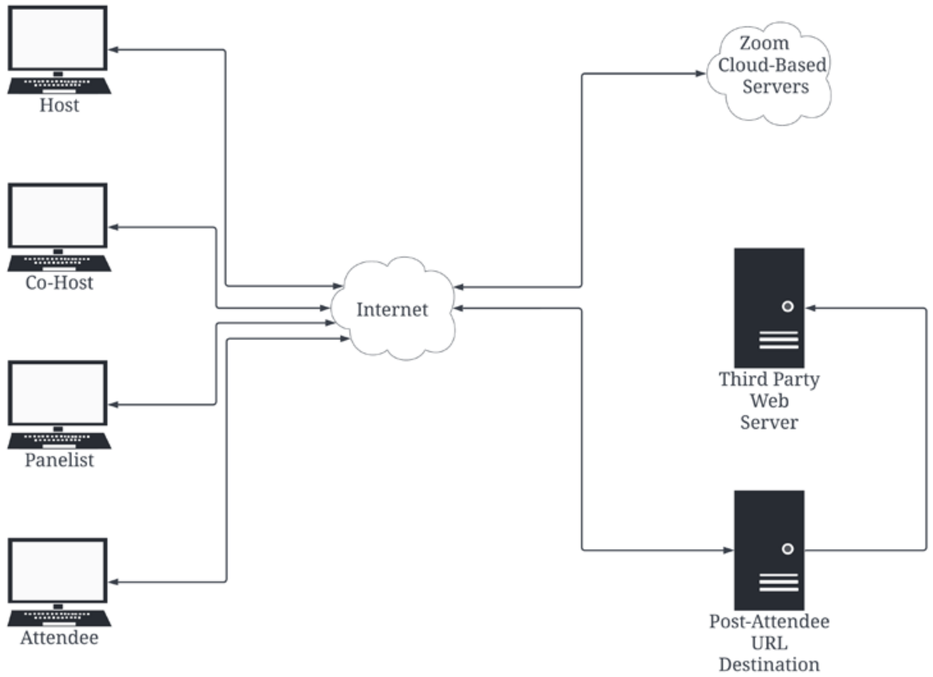


Figure 3: Visual Representation of the Experimental Environment

During these trials, the independent variable, the value that was set as the post-attendee URL, was manipulated to determine if Zoom webinar participants were redirected to malicious content solely due to changing the value of the post-attendee URL. As such, it was important that extraneous variables were identified to avoid altering the reported data in a meaningful way. During the planning stage of this study, the following extraneous variables were identified: the available user roles, the type of device each participant used, the operating system, the default browser, other webinar settings aside from the post-attendee URL, and the version of Zoom that was utilized. However, after conducting the second trial, an additional extraneous variable was identified: the method in which participants joined the webinar.

For the purposes of capturing as much data as possible, live system acquisition techniques such as capturing the data from volatile memory were considered in addition to the forensic images of the desktop hard drives that were created. To minimize the amount of data that was written on the desktops, all external tools for the purpose of live system data acquisition were installed and executed on a 32 GB thumb drive. Additionally, all output files that were generated from these live acquisition tools were stored on the selfsame 32 GB thumb drive for the purpose of minimizing alterations to the Dell OptiPlex desktops.

As mentioned in the previous section, the forensic duplicators were used to create the forensic images throughout this experiment. However, once the desktops were powered off, any data that was stored in the RAM would have been immediately erased. To prevent that data from disappearing, FTK Imager was used to create a memory capture file that contained the contents of the RAM at the time of the capture. Immediately after the webinar was ended by the host in each trial, FTK Imager was executed and captured the contents from the RAM of the Dell OptiPlex desktops into a .mem file, which were later opened by FTK and Magnet AXIOM for further analysis.

Results

Throughout this experiment, four trials involving four unique webinars were created with default settings sans the post-attendee URL. Each webinar was interacted with by four Windows 10 desktops that contained the Zoom desktop application installed. Each Windows 10 desktop had one of the Zoom user accounts from *Table 1* signed in to the Zoom desktop application. In total, 24 forensic images were created. During each webinar, each webinar participant was scored on if the post-attendee URL executed after joining the webinar for five minutes, with each user role scored as “Yes” if the post-attendee URL did execute or “No” if the post-attendee URL did not execute.

Trial 1 Results

Trial 1 was conducted by setting the post-attendee URL to a trusted website. The host logged into the Zoom management portal through the web browser to start the meeting, which as a result caused the host to not join the webinar via email. All participants aside from the host were redirected to the post-attendee URL within five minutes of joining the webinar. See *Table 2* for a summary of the results for this trial.

TABLE 2

Summary of Trial 1 Results

User Role	Zoom Join Method	Did Post-Attendee URL Execute	Browser	Zoom Version
Host	Zoom Web Login	No	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Email Link	Yes	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

Trial 2 Results

Trial 2 was conducted by setting the post-attendee URL as www.bunny meadow.com, a malicious website that the author of this study created which contained a JavaScript file that included a keylogger that transmitted keystrokes on the malicious webpage to a third-party server. For trial 2, the host logged in via email as the other participants. However, the panelist participant never received an email invitation to join the Webinar. As a result, the panelist joined by launching the Zoom application manually and entering the meeting ID and password into the app. The host, co-host, and attendee were redirected to the post-attendee URL, but the panelist was not. For the participants that were redirected to the post-attendee URL, a test was conducted to determine if the keylogger was functioning by asking the participants to type in a random word onto the webpage. The keylogger worked on all three participants who were redirected to the post attendee URL. See *Table 3* for a summary of the results for this trial.

TABLE 3

Summary of Trial 2 Results

User Role	**Zoom Join Method**	**Did Post-Attendee URL Execute**	**Browser**	**Zoom Version**
Host	Email	Yes	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Zoom Desktop Application	No	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

Trial 3 Results

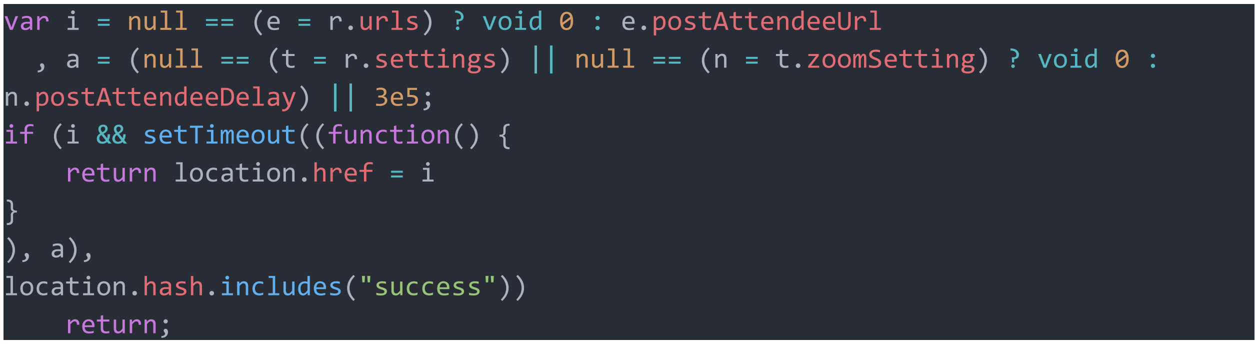
Since the independent variable and an extraneous variable were altered in the second trial, trial 3 was conducted to replicate the results from trial 2 in order to confirm that the joining method of the panelist was the cause for the panelist participant to be exempt from the post-attendee URL redirection. The same URL was used as the post-attendee URL as was used in trial 2. Once all four participants joined the webinar via the invitation email link, all four participants were redirected to the post-attendee URL. To verify that the keylogger still functioned correctly for the participants, the same test from trial 2, only for this trial a different secret word was used. The keylogger successfully transmitted the data to the third-party server amongst all four webinar participants. See *Table 4* for a summary of the results from this trial.

TABLE 4

Summary of Trial 3 Results

User Role	Zoom Join Method	Did Post-Attendee URL Execute	Browser	Zoom Version
Host	Email Link	Yes	Microsoft Edge	5.7.5
Co-Host	Email Link	Yes	Microsoft Edge	5.7.5
Panelist	Email Link	Yes	Microsoft Edge	5.7.5
Attendee	Email Link	Yes	Microsoft Edge	5.7.5

In addition to the data collected in *Table 4*, manual web scraping was performed on the webpage that appeared after clicking on the email invitation link for the host and attendee to determine if the post-attendee URL launch instructions could be discovered. This method uncovered a JavaScript file with over 12,000 lines of code that was found on both the host and attendee post-invitation webpage labeled *meeting.<unique_identifier>.js*, with the unique identifier containing 20 hexadecimal characters. Initial examination of the file uncovered variable names with suggestive labels such as "postAttendeeUrl" and "postAttendeeDelay." An excerpt of the source code from this file can be seen in Figure 4.



```
var i = null == (e = r.urls) ? void 0 : e.postAttendeeUrl
  , a = (null == (t = r.settings) || null == (n = t.zoomSetting) ? void 0 :
n.postAttendeeDelay) || 3e5;
if (i && setTimeout((function() {
    return location.href = i
}
), a),
location.hash.includes("success"))
    return;
```

Figure 4*: Excerpt From meeting.<unique_identifier>.js (Lines 6,394 – 6,401)*

Trial 4 Results

Trial 4 was conducted by setting the post-attendee URL to a direct download link of a JPEG file. Unlike the previous URLs that redirected to a specific website, this URL, when executed, downloads a JPEG file that is stored on the author's Google Drive. All four participants joined the webinar via the invitation email link and were redirected to the post-attendee URL five minutes after joining the webinar. However, instead of going to a specific website, the download process for the JPEG file started and was placed into the default download directory for each device. It's important to note that during this Trial, the Zoom desktop application forced an update that updated the client from version 5.7.5 to 5.9.3. Unfortunately, the client was unable to downgrade back to version 5.7.5. After reviewing the Windows release notes from Zoom's website, the authors determined that the post-attendee URL was not updated and remained unchanged. Nevertheless, precautions were taken and the JavaScript file from the

Zoom launch page was collected again to compare it to the version that was collected in trial 3. Based on the comparison and the results from trial 4, it was determined that the post-attendee URL was unaffected by the update to version 5.9.3. See *Table 5* for a summary of the results from this trial.

TABLE 5

Summary of Trial 5 Results

User Role	**Zoom Join Method**	**Did Post-Attendee URL Execute**	**Browser**	**Zoom Version**
Host	Zoom Web Login	No	Microsoft Edge	5.9.3
Co-Host	Email Link	Yes	Microsoft Edge	5.9.3
Panelist	Email Link	Yes	Microsoft Edge	5.9.3
Attendee	Email Link	Yes	Microsoft Edge	5.9.3

Discussion

In our review of the Zoom documentation and with our experimental trials we demonstrated that there are currently little, if any, security protocols that are in-place to prevent a Zoom webinar hosts to utilize the post-attendee URL feature in bad faith by either sending webinar participants to compromised websites or forcing webinar participants to download malicious files. However, from the examination of the

JavaScript file with over 12,000 lines of code that was recovered in the third trial, there is evidence that suggest that the post-attendee URL can only be executed when Zoom's launch page is opened after webinar participants join via the email invitation link. This is further corroborated with the results from the first two trials, as the only time the post-attendee URL failed to execute was when a webinar participant joined via a means other than the email invitation link. In other words, webinar participants who join a webinar manually from the Zoom desktop application by entering the meeting ID and passcode, when applicable, rather than clicking on the email invitation link directly will not be redirected to the post-attendee URL even if a post-attendee URL is set by a host.

The evidence also suggests that users can prevent the post-attendee URL from executing even if they joined by the email invitation link by closing the Zoom launch page after joining the webinar. Referring once again to JavaScript file collected from trial 3, it is clear that this script uses the JavaScript function *setTimeout()* to create a five minute time once the webinar participant joins the webinar. Once that five minute timer reaches zero, the variable *location.href*, which when assigned a value is treated as a command to navigate to the provided URL, is assigned the value of the post-attendee URL. This modus operandi is consistent with how the public documentation provided by Zoom about the post-attendee URL. However, by knowing that the five minute timer is called by the *setTimeout()* function, this means that the timer is not persistent whenever the state of the Zoom launch page is altered. In other words, if the Zoom launch page was closed, the process responsible for the *setTimeout()* process would be killed as well and the post-attendee URL will never execute, as long as the process was killed before the timer reached zero.

Regarding the digital forensic artifacts that are generated from the post- attendee URL, the “History” database file, as mentioned in the previous section, will provide evidence of the post-attendee URL occurring since a record will be created in that database file that indicates that the web browser transitioned from the Zoom launch page to the value of the post- attendee URL that was set by the host. Timestamps are also stored when a new record is created, which can be pivotal to a digital forensic investigator when creating a timeline of events.

Unfortunately, the Zoom database files that were mentioned in the 2021 paper published by Mahr et al. are now encrypted and the data contained within those files are unavailable to be examined without knowledge of the decryption keys. The memory capture was analyzed as an attempt to retrieve any of the keys in case they were stored in the RAM. Unfortunately, these keys were unable to be recovered.

Throughout this study, several limitations arose that need to be addressed. Perhaps the most glaring limitation of this study was the choice to only use Microsoft Edge throughout this experiment. The reasoning behind this choice is twofold; first, Microsoft Edge comes preinstalled on modern Windows 10 desktops. By using Microsoft Edge, this allowed the baseline forensic images to be close to a fresh installation of a Windows 10 operating system, which minimized the number of software installations. Additionally, Microsoft Edge was chosen since it is a chromium-based browser. As a chromium-based browser, Microsoft Edge shares multiple characteristics with other chromium-based browsers, including how the “History” database file is stored. This would allow the result of the forensic examination to become pertinent to multiple web browsers. However, there is a case to be made that this study excludes non-chromium-based web browsers in the sense that the results may not be as relevant as it wouldbe for chromium-based browsers. Another evident limitation of this study was the use of only Windows 10 desktops without testing the post-attendee URL on other devices, such as Android and iOS devices, and other operating systems, such as Macs or Linux devices. This was mostly due to budgetary concerns, as the only type of experimental devices on-hand were Windows 10 desktops.

Conclusion

Throughout the conducted trials, it is evident that the post-attendee URL contains multiple weaknesses that leave it vulnerable to exploitation by bad actors. From sending Zoom webinar participants to a malicious website to forcing a file to download on a participant’s device, there are a lack of security protocols in place if a Zoom webinar host is acting in bad faith when creating the webinar and abusing the post-attendee URL feature. Nevertheless, there are actions that webinar participants can take to mitigate this risk by avoiding the use of the email invitation link and instead manually launch the Zoom desktop application and enter the meeting ID and password. Joining a Zoom webinar with this method will avoid launching the Zoom launch page, which contains the JavaScript code that is responsible for redirecting participants to the post-attendee URL.

In addition to illustrating how the post-attendee URL feature can be exploited, this study also looked at some of the digital artifacts that were created to construct a timeline of events post-incident response. For this purpose, the “History” database file was analyzed due to its ability to maintain records that indicate how a user was directed to a particular URL. Additionally, JavaScript files from the Zoom launch page were analyzed to deconstruct how the post-attendee URL function behaves at a low level. It is important that digital forensics examiners are aware that the post attendee URL can be exploited so that user claims can be corroborated or rejected.

While this study successfully completed its objective of answering all three research questions that were proposed, there is still more information to the post-attendee URL that can be gleamed. For instance, this study did not investigate how the post-attendee URL functioned on other devices and operating systems outside of the Windows 10 desktops, nor did it analyze the digital artifacts generated by the post-attendee URL feature on non-chromium-based browsers. While this study is important in understanding how the post-attendee URL feature functions and how it can be exploited, it is imperative that future research focus on changes to current technologies and on emerging technologies encountered by digital forensics practitioners.

References

[1] Setera, K, FBI Warns of Teleconferencing and Online Classroom Hijacking During COVID-19 Pandemic [Internet]. Federal Bureau of Investigations; 2020 March 30. Available from: https://www.fbi.gov/contact-us/field-offices/boston/news/press-releases/fbi-warns-of-teleconferencing-and-online-classroom-hijacking-during-covid-19-pandemic

[2] FBI Warns of Child Sexual Abuse Material Being Displayed During Zoom Meetings [Internet]. Federal Bureau of Investigation; 2020 May 20. Available from: https://www.fbi.gov/news/pressrel/press-releases/fbi-warns-of-child-sexual-abuse-material-being-displayed-during-zoom-meetings

[3] Post-attendee URL [Internet]. Zoom Video Communications, Inc; [updated 2021 December 31]. Available from: https://support.zoom.us/hc/en-us/articles/360000518526-Post-attendee-URL

[4] Mahr A, Cichon M, Mateo S, Grajeda C, Baggili I. Zooming into the pandemic! A forensic analysis of the Zoom Application. Forensic Science International: Digital Investigation. 2021;36. https://doi.org/10.1016/j.fsidi.2021.301107

[5] Nicoletti M, Bernaschi M. Forensic analysis of Microsoft Skype for Business. Digital Investigation. 2019;29:159–179. https://doi.org/10.1016/j.diin.2019.03.012

[6] Azab A, Watters P, Layton R. Characterizing Network Traffic for Skype Forensics. Third Cybercrime and Trustworthy Computing Workshop. 2012;p. 19–27. https://doi.org/10.1109/CTC.2012.14

[7] Zoom Connection Process [Internet]. Zoom Video Communications, Inc; 2020 April. Available from: https://zoom.us/docs/doc/Zoom%20Connection%20Process%20Whitepaper.pdf

[8] Roles in a Webinar [Internet]. Zoom Video Communications, Inc; [updated 2022 May 19]. Available from: https://support.zoom.us/hc/en-us/articles/360000252726-Roles-in-Zoom-Webinars