Cyber threat analysis on online learning and its mitigation techniques amid Covid-19

Cyber Threat Analysis on Online Learning and Its Mitigation Techniques Amid Covid-19

[1]Nauman Nazar
School of Computing and Eng
University of West London
United Kingdom
21407254@student.uwl.ac.uk

[2] Iman Darvishi
School of Computing and Eng
University of West London
United Kingdom
21488578@student.uwl.ac.uk

[1]Abel Yeboah-Ofori
School of Computing and Eng
University of West London
United Kingdom
abel.yeboah-ofori@uwl.ac.uk

***Abstract*—The impact of the COVID-19 pandemic on face-to-face teaching and learning affected the world, leading to physical and psychological health issues, especially for the visually impaired and autistic users, including threats to healthcare, economies, and education sectors. During this challenging period, online learning and educational tools such as Zoom, Google Meet, Microsoft Teams, and Cisco Webex gained immense popularity in academic institutions. However, these tools provided vulnerabilities for malicious attackers to exploit these online platforms. That posed a cyber threat to the online educational system to continue and survive under such circumstances. The paper aims to explore and analyze the cyber threats to these online learning platforms to understand the security posture and mitigation techniques. The contribution of this paper is threefold: First, we explore the various attacks on online tools such as Zoom, Google Meet, Microsoft Teams, and Cisco Webex and determine how much security and privacy they offer. Secondly, we analyze the encryption capabilities to assess the confidentiality, integrity, and availability level they provide to the users and present the results as a table. Finally, we discussed a common vulnerability framework comprising common threats faced by users and the service provider for the mitigation techniques to improve security.**

Keywords:* *Cyber Threat Analysis, Covid-19, Cyberattack, Online Learning, Mitigation Techniques

I. INTRODUCTION

The impact of Covid-19 on educational institutions and people with disabilities has been phenomenal as the change to remote and distance teaching and learning approaches have transformed completely, leading to physical and psychological health issues, especially for the visually impaired and autistic users [1] [2] [3]. The transition from face-to-face teaching to remote or online teaching and learning has affected collaborative teaching, and institutions have faced massive difficulty in managing the operations of the learning management systems. Recent developments in research and innovations in adaptive intelligent teaching systems (ITS) [1] and smart learning approaches, including intelligent tutor collaborative learning (ITCL) [4] and group formation platforms, have enhanced computer-supported collaborative learning approaches to assist people with disabilities to study at the higher educational levels.

Most critical domains affected by the impact of Covid-19 include economic, healthcare systems, financial services, media, social, fiscal and educational institutions, as they all have one thing in common: cyber operations [5]. Online educational tools like Zoom, Google Meet, Microsoft Teams, and Cisco Webex assist in flexible learning to improve readily available online learning educational tools and support the universities to sustain their operations. Academics and students could not enjoy physical classroom learning due to the strict lockdown and COVID-19 SOPs in many parts of the world, leading them to the paradigm shift towards online education. These institutes used online educational tools like Zoom, Google Meet, Microsoft Teams, and Cisco Webex, along with already implemented learning management systems [6]. Figure 1 indicates a tenfold increase on these platforms, and most educational users chose Zoom as their primary tool for online education for two reasons. First, it provides free usage of 45 minutes with the facility of multiple users that can enter the meeting without having the challenge of creating their zoom account, and only the meeting host needs to have their account. And the second reason is the ease of access as it has a straightforward user interface.

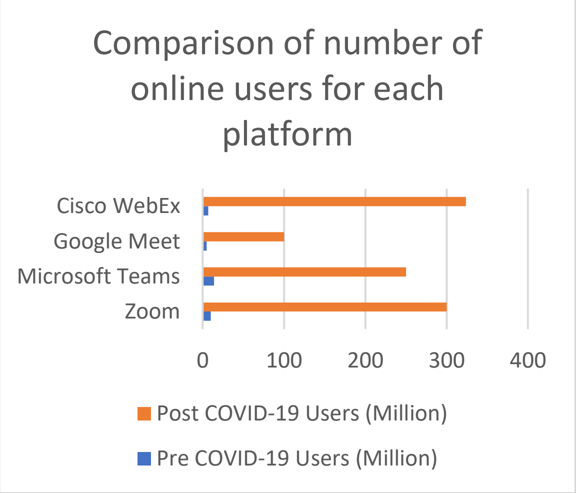


Fig 1. Growth Number of Users Per Day

The current shift towards these online tools presented massive pressure on the software developers to use agile methods to increase business opportunities for their products by increasing features to support online learning. However, these features resulted in software vulnerabilities.

The paper aims to explore and analyze the cyber threats to these online learning platforms to understand the security posture and mitigation techniques. The contribution of this paper is threefold: First, we explore the various ITS and attacks on online tools such as Zoom, Google Meet, Microsoft Teams, and Cisco Webex and determine how much security and privacy they offer. Secondly, we analyze the encryption capabilities to assess the level of confidentiality, integrity, and availability they provide to the users and present the results in the form of a table. Finally, a discussed common vulnerability framework comprising common threats faced by users and service providers was for mitigation techniques to improve security.

II. RELATED WORKS

This section discusses the start-of-the-art and the related works in ITS, online learning platforms and its impact on visually impaired users.

Regarding the importance of video conference technology for education [8] highlighted the relevance of the video conference platforms and revealed how they provide a global platform for other fields. Singh et al. conducted a parametrized comparative analysis of performance between proposed adaptive and personalized tutoring by evaluating existing platforms such as Moodle, Course Builder and Teachable tutoring systems to develop a proposed intelligent tutoring system [1]. Further, Haq et al. proposed a novel intelligent tutor collaborative learning approach for next-generation dynamic group formation learning by using learning patterns and knowledge levels to form a group. Then use heterogeneous balanced groups to augment the collaborative learning for the intelligent tutoring system [4]. A paired T-Test analysis was applied statistically to evaluate the recorded observations. Khattak et al. presented a novel WLAA RSS-Based fingerprinting for indoor localization using a machine learning technique and bag-of-features approach on a k-nearest neighbour classifier to categorize the frequency of vocabulary in a smart educational environment [7]. Rehman et al. review mobile app features for people with ASD in a post-covid-19 era and how it impacts their wellbeing. The authors downloaded and analyzed apps based on eye tracking, facial expression analysis, haptic feedback, and text speech for applied behaviour analysis therapy to assist healthcare professionals in designing future support tools [2]. Nasralla, proposed an innovative JavaScript-based interactive framework with backtrack algorithms for online teaching for students to observe the step-by-step implementation when executed [9] geographically. Khattak et al. proposed a WLAN access point channel assignment strategy for indoor localization systems in smart, sustainable cities by using a technique that ensures the proposed AP channel assignment algorithm in the network scheme mitigates interference in figure prints in crowded areas [10].

A researcher by Video Conferencing Technology and Risk [11] posits that about six types of risks associated with online conferencing and collaboration technologies risk during the software development life cycle. That includes personal information leakage risk, data interception risk during transit, stored data access violation risk, personal privacy risk, and influence operation risk. Google Zero Project. Isobe and Ito, evaluate Zoom's end-to-end encryption scheme by demonstrating three impersonation attack methods based on no entity authentication. Every Zoom user impersonates the user if it has a shared device [13]. Marczak and Scott, examined how Zoom used its encryption protocol for data transmission between users and the Zoom server. However, it is not considered an industry-standard practice [14].

All the related works are relevant and contribute to recent Smart cities and ITS innovations. However, none used Kali Linux to compromise the online platform during collaborative teaching and learning.

III. APPROACH

This section considers the qualitative, subjective approach for the paper as it is impossible to quantify the human behaviour and impact of the attack process. Therefore, a subjective judgement was applied to analyze these educational tools' threats. We used multiple techniques and tools to test the security of systems and then rank them according to the cyber security paradigm. Further, we evaluated the security posture of the online education supporting tools that were common among educational institutes. Zoom, Meet, MS Teams and Webex were chosen based on their consumer base and market share, so the research should be relatively valid and applicable. However, these tools were selected due to constraints like time and resources. For the implementation, we develop a lab environment using real-time calls between users and simulate actual meetings with the help of the virtualization technique.

The design and implementation are divided into two parts. The first part is the dynamic analysis of online education tools using a lab environment. We connect virtual machines, operating systems, target software executables, packet capturing, traffic analysis software, and websites. It results in validation of security and privacy claims made by the platform owners and highlights any security concerns present in the tool. Secondly, the vulnerabilities were evaluated for the platforms. We adopted the MITRE security framework and the Common Vulnerabilities and Exploitation (CVE) [15] to assess the impact of the vulnerabilities identified.

All the virtual machines, including TeacherVM, StudentVM, and AttackerVM configured with NAT network setting to use the main host computer network adapter to connect to the open internet. We used a MAC operating system with up-to-date builds and all desktop applications for the implementation and the latest versions for the 64-bit windows system. However, the selection of operating systems does not affect the research result as the primary target of the paper is focusing on desktop and web applications.

IV. IMPLEMENTATION

This section discusses the infrastructure used to set up the virtual environment, the implementation process, the tools used, and the MITM attack deployed.

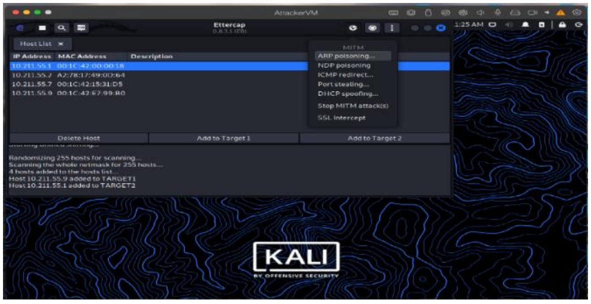
A. Infrastructure Setup

The infrastructure used for the implementation includes operating systems, software, and hardware to analyze and verify security claims by online education tools selected for this research. All the virtual machines, including TeacherVM, StudentVM, and AttackerVM have been configured with NAT network setting to use the main host computer network adapter to connect to the open internet. At the time of implementation, all operating systems were latest with up-to-date builds, and all desktop applications were of the latest versions for the 64-bit windows system. The selection of operating systems does not affect the research result as the primary target of the paper is focusing on desktop applications and web applications.

B. Implementation Process

The implementation process in this research involves different phases. In the first virtualization phase, a virtualization software, Parallels Desktop, was installed on the host Mac machine. Next, three virtual machines, including Windows and Linux machines, were created and configured such that they can communicate with each other to test, capture, and analyze the security of online educational software.

The next phase involves downloading and installing all the online educational platforms from the internet onto those virtual machines. Finally, the last phase consists of the demonstration of man in the middle attack by ARP poising, actual meeting call implementation between endpoints, and capturing all the meeting packets for further analysis by Wireshark.



```
IP Address   MAC Address        Description
10.211.55.1  00:1C:42:00:00:18
10.211.55.2  A2:78:17:49:00:64
10.211.55.7  00:1C:42:15:31:D5
10.211.55.9  00:1C:42:E7:99:B0
```

```
Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
4 hosts added to the hosts list...
Host 10.211.55.9 added to TARGET1
Host 10.211.55.1 added to TARGET2
```

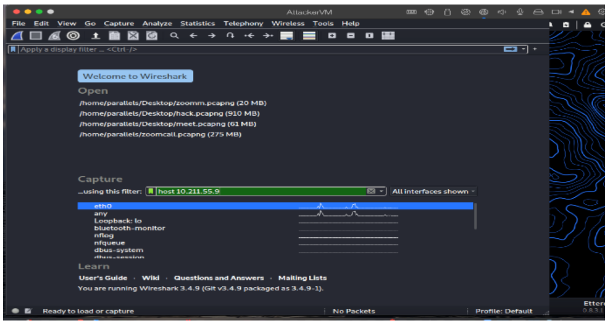
Fig 2. Adding the Target Device to Ettercap

C. Man In The Middle Attack

We used A man-in-the-middle attack for the implementation to assess the encryption of online educational software tools. This attack ensures the capture of all communication traffic by two nodes on the network by the third malicious node. Ettercap application was used to perform this attack via Address Resolution Protocol (ARP) poisoning attack. ARP protocol is used to resolve the MAC address of a host, given that we know about the IP address of that host. Therefore, AttackerVM will be using ARP packets to poison the cache of TeacherVM by sending him ARP packets that its default gateway 10.211.55. Figure 3 shows how the MAC address of AttackerVM MAC address and Ettercap also generates the ARP packets to the default gateway, poisoning its cache to believe that the IP Address 10.211.55.9 has the MAC address of AttackerVM MAC Address.

D. Network Traffic Capturing

Figure 4 shows the same Kali Linux machine was used to capture the traffic between victim machines using the Wireshark application, installed by default on Kali Linux. Setting up a capture filter in Wireshark for a specific target machine IP address reduces the captured scope to only victim machine network traffic. A test case Scenario Zoom meeting call was set up between two virtual machines with default settings. A call was initiated on the TeacherVM, and Meeting ID was created by Zoom with the passcode while StudentVM joined the call with the help of a shared link. The meeting was joined by bothVMs using audio and video features enabled. Test messages were exchanged by using the chat feature provided by the Zoom application to be captured by Wireshark. We tried to upload an executable file in the chat file attachment option, but it failed to load the file by the given extension (.exe). It comes up with an error saying, "File is blocked for security reasons. File Name: "goodfile.exe". TCPView information was captured to extract the IP addresses used by the Zoom application during the meeting for its audio, video, chat, and file data. All the statistics were also captured and saved with the help of the Netstat tool on victim machines. Once all the communication traffic is captured on the attacker's VM machine, the call is terminated.


AttackerVM
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Open
Capture
All interfaces shown
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists
Ettercap

```
/home/parallels/Desktop/zoomm.pcapng (20 MB)
/home/parallels/Desktop/hack.pcapng (910 MB)
/home/parallels/Desktop/meet.pcapng (61 MB)
/home/parallels/zoomcall.pcapng (275 MB)
```

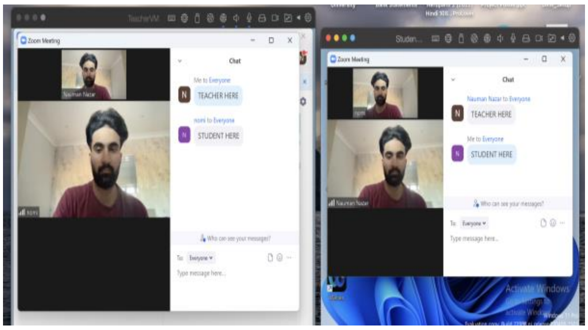
...using this filter: host 10.211.55.9

```
eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
```

You are running Wireshark 3.4.9 (Git v3.4.9 packaged as 3.4.9-1).

Ready to load or capture | No Packets | Profile: Default

Fig 3. Network traffic capture on attack



Me to Everyone: TEACHER HERE

nomi to Everyone: STUDENT HERE

Nauman Nazar to Everyone: TEACHER HERE

Me to Everyone: STUDENT HERE

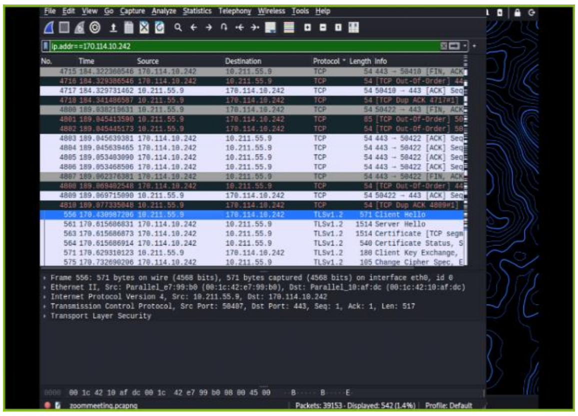
Fig 4. Zoom call setup using the virtual environment

The exact process was repeated for every software application in the same sequence of steps, including Cisco Webex, Google Meet, and Microsoft Teams. Results were analyzed in the form of pcapng format files generated by Wireshark.

E. Analysis

To perform the analysis, Zoom's IP address was extracted from TCPView to verify the encryption and security. Then, the display filter feature provided by Wireshark with the IP Address 170.114.10.242 was used to find all packets transmitted. It was the first IP address in the stream sees Figure 5.

The same analysis was performed for all other software applications. The pcapng files were analyzed with the help of Wireshark display filters, shortlisting IP addresses using TCPView and netstat results. WHOIS Lookup website was also used to check all the IP addresses found by netstat and TCPView to look for their hostnames and geographic locations to validate that they are not sending any data to different.


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==170.114.10.242

```
No.  Time           Source          Destination     Protocol  Length Info
4715 184.322360546 170.114.10.242  10.211.55.9     TCP         54 443 → 50410 [FIN, ACK
4716 184.329386546 170.114.10.242  10.211.55.9     TCP         54 [TCP Out-Of-Order] 44
4717 184.329731462 10.211.55.9     170.114.10.242  TCP         54 50410 → 443 [ACK] Seq
4718 184.341486587 10.211.55.9     170.114.10.242  TCP         54 [TCP Dup ACK 4717#1]
4800 189.038219631 10.211.55.9     170.114.10.242  TCP         54 50422 → 443 [FIN, ACK
4801 189.045413590 10.211.55.9     170.114.10.242  TCP         85 [TCP Out-Of-Order] 50
4802 189.045445173 10.211.55.9     170.114.10.242  TCP         54 [TCP Out-Of-Order] 50
4803 189.045639381 170.114.10.242  10.211.55.9     TCP         54 443 → 50422 [ACK] Seq
4804 189.045639465 170.114.10.242  10.211.55.9     TCP         54 443 → 50422 [ACK] Seq
4805 189.053403090 170.114.10.242  10.211.55.9     TCP         54 443 → 50422 [ACK] Seq
4806 189.053468506 170.114.10.242  10.211.55.9     TCP         54 443 → 50422 [ACK] Seq
4807 189.062376381 170.114.10.242  10.211.55.9     TCP         54 443 → 50422 [FIN, ACK
4808 189.069402548 170.114.10.242  10.211.55.9     TCP         54 [TCP Out-Of-Order] 44
4809 189.069715090 10.211.55.9     170.114.10.242  TCP         54 50422 → 443 [ACK] Seq
4810 189.077335048 10.211.55.9     170.114.10.242  TCP         54 [TCP Dup ACK 4809#1]
 556 170.430987206 10.211.55.9     170.114.10.242  TLSv1.2    571 Client Hello
 561 170.615686831 170.114.10.242  10.211.55.9     TLSv1.2   1514 Server Hello
 563 170.615686873 170.114.10.242  10.211.55.9     TLSv1.2   1514 Certificate [TCP segm
 564 170.615686914 170.114.10.242  10.211.55.9     TLSv1.2    540 Certificate Status, S
 571 170.629310123 10.211.55.9     170.114.10.242  TLSv1.2    180 Client Key Exchange,
 575 170.732690206 170.114.10.242  10.211.55.9     TLSv1.2    105 Change Cipher Spec, E
```

```
› Frame 556: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits) on interface eth0, id 0
› Ethernet II, Src: Parallel_e7:99:b0 (00:1c:42:e7:99:b0), Dst: Parallel_10:af:dc (00:1c:42:10:af:dc)
› Internet Protocol Version 4, Src: 10.211.55.9, Dst: 170.114.10.242
› Transmission Control Protocol, Src Port: 50407, Dst Port: 443, Seq: 1, Ack: 1, Len: 517
› Transport Layer Security
```

```
0000  00 1c 42 10 af dc 00 1c  42 e7 99 b0 08 00 45 00   ··B·····B·····E·
```

zoommeeting.pcapng | Packets: 39153 · Displayed: 542 (1.4%) | Profile: Default

Fig 5. Zoom IP Address Traffic Capture

F. *Threat Analysis*

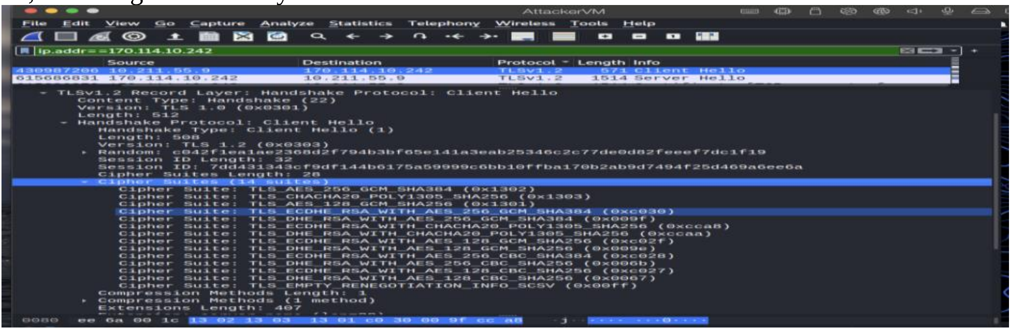
This paper considers multiple threats with their associated vulnerabilities and mitigation techniques divided into the following categories. Security Configuration and, in some cases, the default security settings of some tools are not secure for the user. User needs to modify the settings manually to make their data security and privacy viable. Many threat actors who know the technicality of those security misconfigurations can exploit your privacy. One example is Zoom which is not shipped with data encryption by default.

G. *Zoom Bombing*

It is the new term used for the event when an unintended user gets into a meeting call or class lecture through the digital platform. The word Zoom in it depicts the Zoom feature, which allows any individual to join in the event by guessing only meeting ID, which consists of merely nine characters. This behaviour is not limited to only Zoom; other tools face the same issue too. If a malicious user successfully enters a classroom, business meeting, or other online video conferences, they get offensive with abusing, hate speech, and even hijacking the screen share feature. There was no evidence for end-to-end encryption from Zoom, Teams, and Google Meet. They all could see the contents of video audio streams because they generated, managed, and distributed the session keys for every participant in the meeting.

However, Cisco Webex provided a different approach to end-to-end security. Webex provides a special meeting code that they claimed was serving as the encryption key and was changed upon entering any user into the meeting. So, a new key was distributed among all the users to encrypt the communication from scratch. Our analysis also supports that end-to-end encryption might happen due to the connection with loopback IP addresses. However, that was not seen as the default behaviour of Webex. Instead, the host of the meeting needed to configure for end-to-end encryption. However, it was impossible to verify their claim due to technical limitations and the non-availability of Webex source code. However, giving them the benefit of the doubt and their previous track record regarding security and privacy, their claim can be accepted for end-to-end encryption.

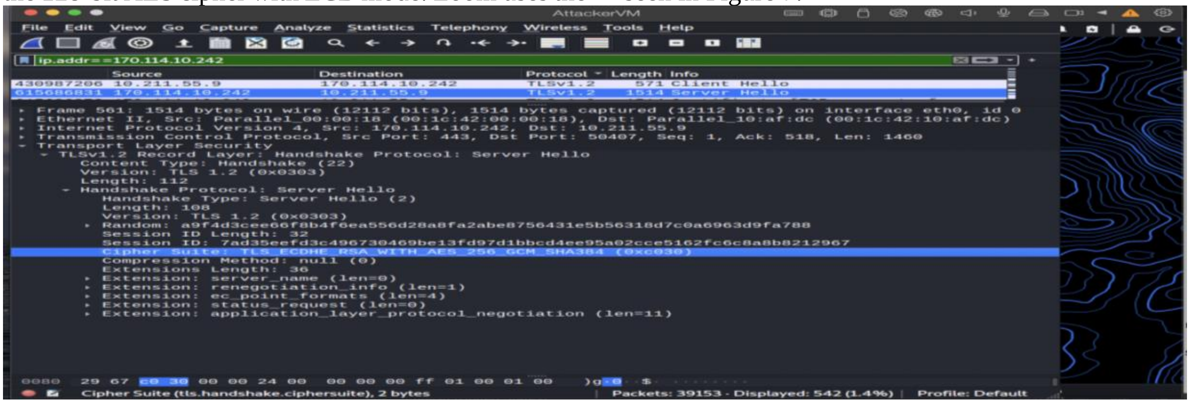
One call was made using the Zoom desktop application from TeacherVM to StudentVM, and all the packets captured by the Wireshark were analyzed manually. Zoom was using 14 different ciphers for its Transport Layer Encryption visible in its server hello packets. Zoom is no longer using its crypto, as visible in Figure 6.



```
ip.addr==170.114.10.242
          Source          Destination       Protocol  Length Info
430987206 10.211.55.9     170.114.10.242    TLSv1.2    571 Client Hello
615686831 170.114.10.242  10.211.55.9       TLSv1.2   1514 Server Hello
  TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 512
    Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 508
      Version: TLS 1.2 (0x0303)
      Random: c042f1ea1ae2368d2f794b3bf65e141a3eab25346c2c77de0d82feeef7dc1f19
      Session ID Length: 32
      Session ID: 7dd431343cf9df144b6175a59999c6bb10ffba170b2ab9d7494f25d469a6ee6a
      Cipher Suites Length: 28
      Cipher Suites (14 suites)
        Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
        Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
        Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0x009f)
        Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
        Cipher Suite: TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xccaa)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (0x009e)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 (0x006b)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
        Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 (0x0067)
        Cipher Suite: TLS_EMPTY_RENEGOTIATION_INFO_SCSV (0x00ff)
      Compression Methods Length: 1
      Compression Methods (1 method)
      Extensions Length: 407
0080  ee 6a 00 1c 13 02 13 03  13 01 c0 30 00 9f cc a8   ·j·· ···· ···0····
```

Fig 6. Zoom client, hello

The cipher used by Zoom is *TLS ECDHE RSA WITH AES 256 GCM SHA384*, which is much stronger than the 128-bit AES cipher with ECB mode. Zoom uses the strongest cipher currently implemented in the industry and can be trusted with data confidentiality. It can be seen in Figure 7.

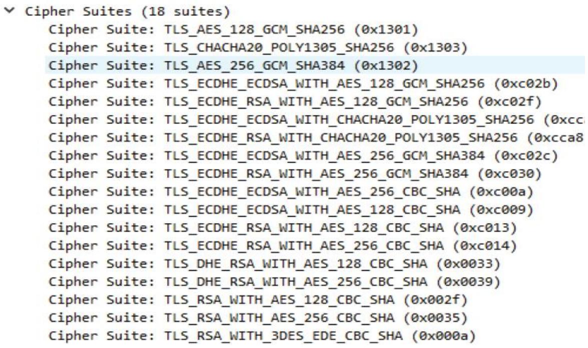


```
ip.addr==170.114.10.242
          Source          Destination       Protocol  Length Info
430987206 10.211.55.9     170.114.10.242    TLSv1.2    571 Client Hello
615686831 170.114.10.242  10.211.55.9       TLSv1.2   1514 Server Hello
Frame 561: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface eth0, id 0
Ethernet II, Src: Parallel_00:00:18 (00:1c:42:00:00:18), Dst: Parallel_10:af:dc (00:1c:42:10:af:dc)
Internet Protocol Version 4, Src: 170.114.10.242, Dst: 10.211.55.9
Transmission Control Protocol, Src Port: 443, Dst Port: 50407, Seq: 1, Ack: 518, Len: 1460
Transport Layer Security
  TLSv1.2 Record Layer: Handshake Protocol: Server Hello
    Content Type: Handshake (22)
    Version: TLS 1.2 (0x0303)
    Length: 112
    Handshake Protocol: Server Hello
      Handshake Type: Server Hello (2)
      Length: 108
      Version: TLS 1.2 (0x0303)
      Random: a9f4d3cee66f8b4f6ea556d28a8fa2abe8756431e5b56318d7c0a6963d9fa788
      Session ID Length: 32
      Session ID: 7ad35eefd3c496730469be13fd97d1bbcd4ee95a02cce5162fc6c8a8b8212967
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
      Compression Method: null (0)
      Extensions Length: 36
      Extension: server_name (len=0)
      Extension: renegotiation_info (len=1)
      Extension: ec_point_formats (len=4)
      Extension: status_request (len=0)
      Extension: application_layer_protocol_negotiation (len=11)

0080  29 67 c0 30 00 00 24 00  00 00 00 ff 01 00 01 00   )g·0··$· ········
Cipher Suite (tls.handshake.ciphersuite), 2 bytes | Packets: 39153 · Displayed: 542 (1.4%) | Profile: Default
```

Fig 7 Zoom server hello

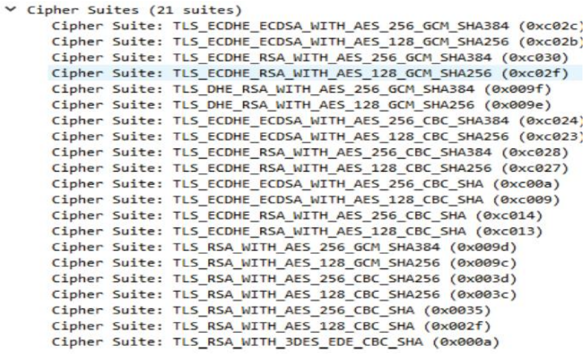
A similar call setup was operated for Google Meet as well. Again, it was observed that Meet provides far more ciphers than Zoom, as seen in Figure 8.



```
▾ Cipher Suites (18 suites)
      Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
      Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
      Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcc
      Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
      Cipher Suite: TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
      Cipher Suite: TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)
      Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
      Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
      Cipher Suite: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000a)
```

Fig 8. Cypher suite used by Google Meet

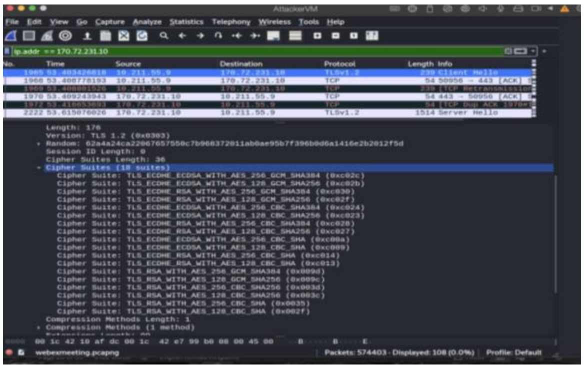
A total of eighteen ciphers were offered in Google Meet, see figure 8, cypher suite. Interestingly It was found that Meet uses AES 128-bit cypher for its encryption leaving out the 256-bit AES cypher. It might be used for efficiency, bandwidth, and delay optimization; however it is relatively weaker than the algorithm used by Zoom. Another tool we analyzed for its encryption strength was Microsoft Teams which was. Evidence shows that it offers more cyphers from Zoom and Meet. Twenty-one offered ciphers can be seen in Figure 9.



```
▾ Cipher Suites (21 suites)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
      Cipher Suite: TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0x009f)
      Cipher Suite: TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (0x009e)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
      Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
      Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
      Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
      Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
      Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA256 (0x003d)
      Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA256 (0x003c)
      Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
      Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
      Cipher Suite: TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000a)
```

Fig 9. Cypher Suite used by Microsoft Teams

These cyphers, which Teams offer, operate in Cipher Block Chaining Mode and Galois/Counter Mode, which is famous for its performance, throughput, and fewer hardware requirements. Any key that is 128-bit AES and 256-bit AES can be selected for encryption. It used TLS *ECDHE RSA WITH AES 256 GCM SHA384* Cipher suite, equaling the security provided by zoom and the strongest cipher in the industry. Cisco Webex uses a strong encryption technique just like Zoom and Microsoft Teams. Its cypher suite is identical to both, as is evident from Figure 10.



AttackerVM
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr == 170.72.231.10
No.   Time          Source         Destination    Protocol  Length Info
1965 53.403426818  10.211.55.9    170.72.231.10  TLSv1.2    239 Client Hello
1968 53.408778193  10.211.55.9    170.72.231.10  TCP         54 50956 → 443 [ACK]
1969 53.408801526  10.211.55.9    170.72.231.10  TCP        239 [TCP Retransmissio
1970 53.409243943  170.72.231.10  10.211.55.9    TCP         54 443 → 50956 [ACK]
1972 53.416653693  170.72.231.10  10.211.55.9    TCP         54 [TCP Dup ACK 1970#
2222 53.615076026  170.72.231.10  10.211.55.9    TLSv1.2   1514 Server Hello
      Length: 176
      Version: TLS 1.2 (0x0303)
    ▸ Random: 62a4a24ca22067657550c7b968372011ab0ae95b7f396b0d6a1416e2b2012f5d
      Session ID Length: 0
      Cipher Suites Length: 36
    ▾ Cipher Suites (18 suites)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 (0xc023)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
        Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
        Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
        Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
        Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA256 (0x003d)
        Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA256 (0x003c)
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
        Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
      Compression Methods Length: 1
    ▸ Compression Methods (1 method)
0000  00 1c 42 10 af dc 00 1c  42 e7 99 b0 08 00 45 00   ..B.....B.....E.
webexmeeting.pcapng                Packets: 574403 · Displayed: 108 (0.0%)   Profile: Default
```

Fig 10. Webex client hello

H. *Security Document Publications*

Zoom has proper documentation regarding its security design. They regularly publish security advisories, patches, and details about new features added to the application. They also update their user if the security or privacy policy is updated.

Google Meet is also up to the mark regarding the security design documentation of their product. They provide compliance reports, transparency documents, and their privacy protection policy regularly upon any update. Their efforts can be seen by looking at their user guide, which highlights best practices for users to stay safe online.

Microsoft Teams does not let itself be left behind by others and provides detailed documents regarding different features, explaining how the features are secure, their product security updates, and security advisories to their customers. It automatically patches its applications without the user being notified, but the users can deactivate the setting.

Webex provides a detailed zero-trust approach toward the security of its customers. It covers all the technical details surrounding their end-to-end encryption service, identity management service, and key management service for better security and user privacy. It also issues security advisories and compliance reports for the public to review its security posture, making it a popular choice for online video collaboration in recent times.

V. DISCUSSIONS AND FINDINGS

All the platforms perform data encryption in the rest, but only Zoom violates this default behaviour. It provides this feature, but the user must configure it manually. However, while it's about storing in the cloud, all platforms encrypt their data stored in the cloud. Regarding end-to-end encryption, only Cisco Webex stands out with its peculiar essential management technique, which empowers them to provide end-to-end encryption. Unfortunately, the design of other tools does not allow them to encrypt the data they cannot access. Furthermore, only Zoom provides the feature of anonymous sign-in to the meeting, which can cause a lot of trouble if the proper security controls are not in place for that meeting, like a waiting room and meeting password. All other platforms require the user to sign in before using their services. A detailed comparison can be seen in Table 1.

TABLE 1. COMPARISON OF SECURITY FEATURES PROVIDED BY ONLINE EDUCATIONAL PLATFORMS

Security Feature	Zoom	Microsoft Teams	Cisco Webex	Google Meet
Encryption During Data Transfer	✓	✓	✓	✓
Encryption at Rest by Default	✗	✓	✓	✓
End-to-End Encryption	✗	✗	✓	✗
User Identity	✗	✓	✓	✓
Documented Security Design	✓	✓	✓	✓
Open Source	✗	✗	✗	✗

Table 2 provides vulnerability analysis used to collect data from the MITRE framework about all the publicly exposed vulnerabilities in these platforms. It contains the data from the pre-COVID-19 era as well. It is summarised in Table 1 with the categorization of vulnerability impact.

TABLE 2. VULNERABILITY COMPARISON WITH THE IMPACT

Platform	High-Risk (CVSS>6)	Medium-Risk (CVSS>=4 & <6)	Low-Risk (CVSS < 4)	Total
Zoom	4	2	3	9
Google Meet	0	0	0	0
Microsoft Teams	1	2	2	5
Cisco Webex	7	2	0	9

Risk Mitigation Guidelines

Table 3 presents security recommendations to be implemented by the users of the online education system and the producers of online educational platforms, that is, owners and developers. These guidelines can be summarized in Table 3

TABLE 3. CYBER RISK MITIGATION SUGGESTIONS

Security Threat	Mitigation for Tool User	Mitigation for Tool Service Provider
Security Configuration	User training on how to use the tool securely.	Enabling Security features by default. Security guidelines and documentation availability.
Zoom Bombing	Creation of meeting with a strong password. Enable waiting room.	Block the users on multiple unsuccessful attempts. Force users to create a meeting with a security password.
Cyber Bullying	Disable the private chat feature. Only enable the one-to-one chat feature.	Use of harassment filter checks and notify any violation to meet the owner in real-time.
Information Correlation	Do not post meeting pictures or other details on social media or web	Options to the user for hiding background
Virus Threats	Keep your digital tools up to date and patched to the latest version. Avoid opening any file received by an untrusted person in the meeting.	And functionality to your application to detect any malicious file uploaded to it by any user and instantly block it and notify the user who tries to upload it and other participants of the meeting that the file is malicious
Leakage of Information	Try to implement virtual background if the video is necessary. Use a pseudo name instead of your real name.	Implement the functionality for the provision of a generic virtual background facility for the video calling feature.
Phishing Campaigns	Create user awareness to validate the website before giving information properly. Users must have a basic level of awareness of SSL and digital signature technologies	Regularly check for any phishing campaign that impersonates your company. For example, check for DNS records if someone registers their website as zoom.net or zoom.org domain or zoomme.com etc.
Enterprise Data Theft	Avoid storing your private and confidential meeting data over the cloud as much as possible.	Implement strong encryption at rest, ensuring that data is encrypted before getting stored on cloud servers.

VI. CONCLUSION

The research aims to analyze online education platforms' security threats and suggest mitigation techniques. For example, zoom was found to have a lazy security approach as it leaves most of the security controls to the application users, which is bad practice as many users are unaware of these settings. Hence, they can compromise their data privacy.

Google Meet has the highest security regarding the reduced surface area, as it only provides a web application for its users and compulsory account sign-in requirement. It also does not have any vulnerability CVE yet due to its only web application strategy, which makes it patch-friendly. But at the same time, it is challenging to use and navigate. On the other hand, Microsoft Teams present a decent security posture managing its platform with the highest level of encryption, User identity management, and data at rest encryption. As a result, it also has the least number of CVE vulnerabilities reported.

Cisco Webex was found to be unique in its end-to-end encryption implementation through its state-of-the-art encryption and key management system. Users need to sign in to prove their identities to join a meeting, and upon joining, it also changes the meeting encryption code for all other users, ensuring that end-to-end encryption is in place.

Finally, some common security threats were identified: viruses, Data Correlation, Phishing, Security configuration, zoom bombing, leakage of information and enterprise data theft, and cyberbullying. These are the most prevalent threats faced by educators over the digital platforms. A comprehensive solution to mitigate or lower the above risks was identified and documented. The guidelines are according to the industry's security standards and require compliance for the users and developers of the online education tools.

Due to time constraints, the paper was confined to only the community's top four tools currently implemented and widely used. Another constraint was the lack of paid subscriptions to the researched platforms. As this research was carried out on the researcher's funding, it was limited to the available free features. Only encryption cyphers and IP addresses with a port number were analyzed in the lab implementation. No other dynamic analysis could be performed due to technical limitations like lack of reverse engineering skills which might be helpful to identify and dig deeper into the analysis of other features. Due to hardware constraints, only desktop and web applications of these platforms were under consideration, leaving out the vast surface area of mobile applications. Finally, vulnerability exploitation was highly difficult as all these software use web applications at the backend to connect to the desktop applications. Even older versions of desktop applications are not available, which are vulnerable to public exploits. Most of the exploits are not public due to disclosure agreements. Which are available, but their target application is not available at this point.

REFERENCES

[1] N. Singh, V. K. Gunjan and M. M. Nasralla, "A Parametrized Comparative Analysis of Performance Between Proposed Adaptive and Personalized Tutoring System “Seis Tutor” With Existing Online Tutoring System," in *IEEE Access*, vol. 10, pp. 39376-39386, 2022, doi: 10.1109/ACCESS.2022.3166261.

[2] I.U. Rehman, D. Sobnath, M. M. Nasralla, M. Winnett, A. Anwar, W. Asif, and H. H. R. Sherazi, “Features of Mobile Apps for People with Autism in a Post COVID-19 Scenario: Current Status and Recommendations for Apps Using A.I. Diagnostics” MDPI. 2021, 11, 1923. https:// doi.org/10.3390/diagnostics11101923a.

[3] M. M. Nasralla, “An Innovative JavaScript-Based Framework for Teaching Backtracking Algorithms Interactively,” *Electronics*, vol. 11, no. 13, p. 2004, Jun. 2022, doi: 10.3390/electronics11132004.

[4] I. U. Haq *et al.*, "Dynamic Group Formation With Intelligent Tutor Collaborative Learning: A Novel Approach for Next Generation Collaboration," in *IEEE Access*, vol. 9, pp. 143406-143422, 2021, doi: 10.1109/ACCESS.2021.3120557.

[5] OECD, “The Territorial Impact of Covid-19: Managing The Crisis Across Levels Of Government” 2020.

[6] T. Warren “The Verge” 2020. https://www.theverge.com/2020/4/30/21242421/zoom-300-million-users-incorrect-meeting-participants-statement

[7] A. Gupta, Role of Video-Conferencing Platforms To Change The Face Of Communication During The Lockdown. DIP: 18.21. 716954795.058

[8] S. B. A. Khattak, M. M. Fawad, M. A. Nasralla, H. Esmail, Mostafa, and M. Jia, “WLAN RSS-Based Fingerprinting for Indoor Localization: A Machine Learning Inspired Bag-of-Features Approach,” *Sensors*, vol. 22, no. 14, p. 5236, Jul. 2022, doi: 10.3390/s22145236.

[9] M. M. Nasralla, “An Innovative JavaScript-Based Framework for Teaching Backtracking Algorithms Interactively,” Electronics, vol. 11, no. 13, p. 2004, doi: 10.3390/electronics11132004

[10] S. B. Khattak, M. M. Nasralla, M. Marey, M. A. Esmail, N. Jia, M. Y. Umair. “WLAN Access Points Channel Assignment Strategy for Indoor Localization Systems in Smart Sustainable Cities.” In IOP Conference Series: Earth and Environmental Science 2022 May 1 (Vol. 1026, No. 1, p. 012043). IOP Publishing.

[11] J. Lewis, “*Video Conferencing Technology and Risk*.” Center for Strategic & International Studies. 2020.

[12] D. Kagan, G. F. Alpert, and M. Fire, “Zooming into video conferencing privacy and security threats.” 2020. *arXiv preprint arXiv:2007.01059*.

[13] T. Isobe and R. Ito, "Security Analysis of End-to-End Encryption for Zoom Meetings," in IEEE Access, vol. 9, pp. 90677-90689, 2021, doi: 10.1109/ACCESS.2021.3091722.

[14] B. Marczak, and J. Scott-Railton, “Move fast and roll your own crypto.” 2020. *Report, The Citizen Lab*.

[15] MITRE Corporation. “Vulnerability in Webex Desktop Apps” 2020 [Online] https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-3263