MITRE ENGENUITY
A Foundation for Public Good

MITRE ENGENUITY TECHNICAL REPORT

Information Security Considerations for Video Collaboration Platforms

Facilitating the Transition to an Increasingly Remote Workforce

Jim Barry
Drew Buttner
Navaneeth Krishnan Subramanian

December 2021

The views, opinions and/or findings contained in this report are those of MITRE Engenuity, Incorporated and should not be construed as an official position, policy, or decision, unless designated by other documentation.

This document has been approved for public release. Document number ME0037

Abstract

Video collaboration platforms are prevalent in today's business environment. An increasingly remote workforce has come to rely on video collaboration software in lieu of the conference room and other shared messaging and file-storage systems that facilitate day-to-day operations. The spike in usage of these tools has occurred almost overnight, and as a result shortcomings in information security have been largely overlooked. This paper presents an overview of the various security issues to consider when a workforce is looking to bolster existing security mechanisms and protocols around these tools or conducting a review of the available collaboration platform options before acquisition.

Table of Contents

1 Introduction

This paper presents various cybersecurity concerns that an organization should be cognizant of before developing or acquiring video collaboration tools. The following sections discuss potential vulnerabilities and/or shortcomings in video collaboration tools and how these can impact an organization’s overall cybersecurity posture.

This paper looks to inform decision makers about policy considerations for video collaboration tools, specifically in the topic areas of storage mechanisms, handling of sensitive data, unauthorized data sharing, authentication of guest users, uploading confidential data, and the challenges of data archiving and auditing. This paper also discusses the impact of using devices outside the corporate infrastructure and bring-your-own-device policies on corporate security posture. Challenges in this area include loss of data, data breaches and leaks through broad-brush authorization, screen sharing, and proximity access to a user’s screen. The paper elaborates on setting encryption standards, authenticating guest users, and considerations around ‘video-on’ collaboration. Finally, the paper proposes mitigation strategies that would be effective in securing and acquiring video collaboration tools for organizational policymaking and workflow.

2 Code Weaknesses

Exploitable vulnerabilities within video collaboration tools originate from software weaknesses, which themselves result from insecure coding practices. If software developers are not properly trained on secure coding best practices, and if proper testing is not conducted (e.g., automated static code analysis and manual code inspection), then a variety of software weakness types will persist past development and exist within a given video collaboration tool's codebase. Legacy code only adds to the challenge, especially if it remains in an unmodified state from a time when many of the security concerns were not fully understood.

This section discusses some of the more common software weakness types that have been identified by MITRE Engenuity during reviews of video collaboration tools. Each weakness area is related to entries within the Common Weakness Enumeration (CWE™), which aims to stop vulnerabilities at the source by educating software and hardware architects, designers, programmers, and acquirers on how to eliminate the most common mistakes before products are delivered. Additional information about CWE[1] can be found at https://cwe.mitre.org.

2.1 Buffer Management

Programming languages that allow developers to directly manipulate data buffers (e.g., C/C++) inevitably have buffer-related weaknesses (see CWE-787: Out-of-bounds Write). These weaknesses most frequently result from arithmetic errors, either due to an incorrect calculation by a developer when determining a buffer's size (see CWE-131: Incorrect Calculation of Buffer Size) or when the language itself must perform casting to ensure types remain consistent (see CWE-843: Access of Resource Using Incompatible Type). In the former case, the developer may allocate an undersized buffer and/or attempt to copy too much data into the buffer. In the latter case, the casted object may change its value to a size larger than the allocated buffer. Buffer-related weaknesses can also manifest when an assumption is made regarding the size of a buffer and the data to be copied, resulting in neither value being checked prior to the manipulative operation. Regardless of how the weakness originates, an adversary could exploit this software weakness to conduct an attack with the goal of reading/modifying memory or executing unauthorized code.

To minimize buffer-related weaknesses in software, developers should ensure that buffers and data sizes align prior to operating on the recipient buffer. Additionally, developers should consider leveraging functions/methods that include safeguards when operating on buffers.

2.2 Resource Management

Resource Management weaknesses are an additional concern within video collaboration tools. Software weaknesses within this category can manifest in several ways, such as the software improperly disposing of resources after their effective lifetimes. This can occur if the software

[1] CWE is sponsored by the U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA) and managed by the Homeland Security Systems Engineering and Development Institute (HSSEDI) which is operated by The MITRE Corporation (MITRE)

does not dispose of the resource (see CWE-772: Missing Release of Resource after Effective Lifetime) or if an unaccounted-for edge case exits that results in the program exiting without proper cleanup (see CWE-460: Improper Cleanup on Thrown Exception). Another origination of the weakness type is the release of an invalid pointer (see CWE-763: Release of Invalid Pointer or Reference). The use of uninitialized resources (see CWE-908: Use of Uninitialized Resource) has also been observed during our reviews.

These types of weaknesses, when found in critical modules of video collaboration software, can ultimately result in unintended read/modification of memory, unauthorized execution of code, or denial of service.

Preventing these weaknesses can be accomplished by leveraging a language's automatic memory-management features, such as Java's "Try with Resources" or C++'s "Resource Acquisition is Initialization" (RAII). Furthermore, developers should ensure resources are properly disposed of in all instances where the resource is no longer needed, this includes the event of an error condition being handled.

2.3 NULL Dereference

Another common weakness within video collaboration tools is Null dereference (see CWE-476: NULL Pointer Dereference). In languages such as C/C++ and Java, Null dereferences can crash an application or cause it to behave in unanticipated manners downstream. Null dereferences can occur if developers assume a value will never be NULL and/or do not check a value for NULL prior to operating on the resource. One instance where this can occur is if an external function returns NULL in the event of an error. A developer may attempt to set some variable 'X' as the return result from said function, while assuming this function always succeeds. Therefore, the developer does not check the return result for NULL prior to operating on 'X'. In the event of an error, 'X' will be dereferenced while NULL, which can result in the application crashing. If an adversary is able to trigger this behavior, it could further result in a denial of service against the application.

If function-return values are stored as variables and later operated on, developers should ensure that all function-return values are checked for NULL prior to proceeding with program execution. Additionally, any value that could be NULL, even in the event of an error, should be checked for NULL prior to operating on the resource.

2.4 Inadequate Encryption

Inadequate encryption strength (see CWE-326: Inadequate Encryption Strength) is a common weakness in video collaboration tools. As the workforce becomes increasingly remote, the importance of securing network communications with strong encryption support cannot be understated.

The underlying software of video collaboration tools should leverage accepted and up-to-date cryptographic protocols, algorithms, and related primitives to ensure that known attacks cannot be successfully executed. If an application leverages cryptography that is known to be insecure, such as 3DES or SHA1, it could open the door for a variety of exploits targeting the confidentiality and integrity of the application and its data.

2.5 Input Validation/Sanitization

Within video collaboration tools, user-driven input typically originates in the form of search queries, chat messages, and uploaded files. If this data is not properly validated and sanitized by the underlying software, it could result in a myriad of attacks including SQL Injection, Cross Site Scripting (XSS), Cross Site Request Forgery (CSRF), Malicious File Upload and Command Injection.

In cases where user input drives the functionality of the software, ensure this data is properly validated and sanitized prior to being operated on. For example, if using SQL, consider leveraging parameterized queries and prepared statements. If file uploading is permitted, ensure only files of necessary types are allowed to be uploaded. For functionalities such as search queries and chat, ensure special characters, such as those leveraged to conduct script-based attacks, are properly escaped to prevent execution of undesired code within the application.

2.6 Third-party Integrations

Third-party integrations, such as libraries, modules, packages, etc., are a final weakness category to consider for video collaboration tools. Although the majority of today's software uses third-party integrations in some form, many organizations do not properly validate their dependencies for trusted pedigree and known vulnerabilities. As a result, these become an enticing attack vector for a would-be adversary. For example, if a software application is using an outdated version of a popular library with known Common Vulnerabilities and Exposures (CVE™), then an adversary could attack the application using this vulnerable library as a launching point.

All third-party integrations should be assessed for pedigree to assure that the library originates from a trusted source. Additionally, third-party components should be assessed for known vulnerabilities and appropriate mitigations should be applied. This usually entails using updated versions of third-party components and not using outdated/deprecated integrations.

3 Data Protection

Organizations have come to rely on video collaboration tools and are entrusting these tools to protect sensitive and important data that is shared within the system. The storage of documents, work products including computer code, software, research materials, and other digital artifacts presents a security challenge.

3.1 External Storage of Information

When working with video collaboration tools, artifacts such as recorded meetings, chat logs, and shared files are common pieces of data processed and stored by the tool. This data may include business secrets, personally identifiable information (PII), or other sensitive information intended only for a specific audience. Where and how this data is stored is an import security factor that organizations must consider before acquiring and implementing one of these tools.

A common approach to data storage is to leverage a centralized location, either internally within the organization or via an external data storage provider, potentially even the tool vendor itself. Data stored externally to an organization can be subject to external protection policies, which may not align with the security objectives of the owning organization. It is important to review and understand the data protection policies used by these external storage locations.

3.2 Unauthorized Data Sharing

Unauthorized data sharing can occur in numerous ways within video collaboration tools and can occur either intentionally or unintentionally. For example, during a meeting a user may disclose information pertaining to a specific project, not knowing that individuals without need-to-know are in attendance. Another example entails a user improperly setting file restrictions for an uploaded file containing sensitive information. A user might also share their screen during a meeting and display information they shouldn’t have or didn’t intend to present.

Organizations leveraging video collaboration tools must ensure users are educated on how to properly restrict data access to individuals who are not authorized to interact with said data. One suggestion is to ensure that proper file restrictions are set for all uploaded files and that uploaded files do not contain sensitive data. An additional recommendation is to only share applications or windows (rather than a user’s desktop) while in sharing mode, in addition to closing out any applications or windows irrelevant to the given discussion. If leveraging a webcam, users should also ensure notes, whiteboards, and other materials are not in sight of other participants. A final suggestion is to avoid text chatting or discussing sensitive or restricted information unless all individuals present possess a need-to-know for the information, recognizing that an audio channel may be “open” prior to the initiation of the video feed.

In addition to data sensitivity considerations, organizations should be prepared to conduct debriefing operations for resources who might have received an artifact above their own clearance level and mitigate the consequences.

Furthermore, conducting audits and assessments of the handling of sensitive and/or confidential data presents a challenge when navigating a digital workspace. The lack of procedures and awareness when it comes to dealing with the handling of these types of information introduces opportunities for compromise, and without clear reporting mechanisms in place, unauthorized information disclosures can go unreported.

3.3 Data Uploads

Data uploaded to video collaboration platforms can also present a security challenge.

One of the biggest concerns is malicious files being uploaded to the tool. For example, an adversary may upload a malicious file to the video collaboration tool with permissions set so that any user can open and/or download this file. If an unsuspecting victim were to open this file, it may result in a code execution attack being conducted on their system. To prevent cases such as this, all files uploaded to video collaboration tools should be assessed for malware prior to being successfully uploaded to the tool.

Video collaboration tools should be configured to properly handle unique or unexpected filetypes. If possible, configure the tool to only allow a known set of file extensions (i.e., an allow list) as this can help prevent potentially malicious files from being uploaded to the tool. Additional filtering should also be implemented to catch attempts at masquerading filetypes.

3.4 Encryption

Cryptographic techniques should be in place to secure communications within the tool. These techniques should be applied to text-based communications in collaboration tool chats, audio-visual communication in video collaboration platforms, as well as command and control channels within the tool. Fundamental standards should include the usage of TLS over HTTPS for encrypted communication, using trusted means for verification of servers within the environment, and enforcing the use of modern standards for encryption and digital signatures.

4 Other Concerns

This section presents some key areas to consider when an organization seeks to harden its security controls and mechanisms around the usage of video collaboration platforms.

4.1 Localized Control to Group Users

Video collaboration platforms sometimes have integrated access controls. A user authorized to access a group on a collaboration platform may have access to the group's digital data on a cloud storage platform developed by the same vendor, access to video conferences on the vendor's video collaboration software, or other available resources. Policy around access controls and the users capable of administering access controls to other users need to be well defined to ensure that a user is not granted access to information outside their privilege level.

4.2 Proximity Access

Attending a video conference is possible from coffee shops, hotels, and other public workspaces. Audio and video from these meetings, can be accessed by someone merely in proximity to an attendee's device, maybe via shoulder surfing or eavesdropping. While this is a difficult scenario to devise security controls for, it is something to inform and train users about.

4.3 Unexpected Attendees

Access to meetings is often controlled by who has the dial-in information or meeting URL. Maybe a meeting password has been established, but true authentication of each attendee is not performed, thus allowing anyone with the meeting information to dial into meetings and video conferences. Unless the host of the meeting has a record of all possible phone numbers that could be dialing in to the meeting and manual-based security procedures to verify authenticity of those numbers, unexpected attendees may be present. This presents concerns of unauthorized personnel gaining access to assumed private meeting discussions. In the day-to-day workings of an organization, authentication of guest users generally ends with a verbal confirmation of the guest's identity. Enforcing tighter control over participation in a meeting from within the tool verses relying on manual procedures would be a more reliable means of controlling access.

4.4 External Devices

As organizations transition to an increasingly flexible, fluid, and "always on" workforce, organizations have been more open to personal devices being used for day-to-day work by its staff. Policies assessed and implemented across the organization to exert security controls over company infrastructure such as laptops, mobile phones, and tablets can be undermined when employees use their personal devices to access their digital workspace or dial into video meetings. This puts the organization's information (including proprietary or confidential information) at greater risk of unauthorized access, as the security barrier is reduced to the security of each employee's device. Stricter policies regarding business use of personal devices and limitations within the application itself could mitigate concerns of this nature.

5 Conclusion

This paper presented various cybersecurity concerns that an organization should be cognizant of before developing or acquiring video collaboration tools. Following industry best-practice, verifying good development procedures, and adhering to well-known security mechanisms, can help mitigate these concerns.

6 References/Bibliography

I. Sotnikov. “4 Collaboration Habits That Open the Door to Security Breaches”. cmswire.com.

https://ieeeauthorcenter.ieee.org/wp-content/uploads/IEEE-Reference-Guide.pdf (accessed Sep. 28, 2021).

E. Kass. “Since The Start Of The Pandemic, Business Use Of Collaboration Tools Has Soared — Bringing New Security And Compliance Challenges”. mimecast.com.

https://www.mimecast.com/blog/collaboration-tools-many-benefits--and-new-security-risks/ (accessed Sep. 28, 2021).

M. Middleton-Leal. “Security risks of increasingly popular cloud collaboration tools”. comparethecloud.net.

https://www.comparethecloud.net/articles/security-risks-of-increasingly-popular-cloud-collaboration-tools/ (accessed Sep. 28, 2021).

Nathan. “Security Risks of using collaboration tools for your Business”. medium.com.

https://medium.com/mongrov/security-risks-of-using-collaboration-tools-for-your-business-4260c2e0f78c (accessed Sep. 28, 2021).