SECURITY TARGET – ZOOM APPLICATION

ZOOM VIDEO COMMUNICATIONS, INC.

Version:

1.8

Date:

15/12/2021

SECURITY TARGET – ZOOM APPLICATION

PREPARED BY

IDENTIFICATION OF SERVICE PROVIDER		
	Organization Name	Zoom Video Communications, Inc.
	Street Address	55 Almaden Boulevard
	Suite/Room/Building	6th Floor
	City, State Zip	San Jose, CA 95113

TABLE OF CONTENTS

INTRODUCTION

ST-REFERENCE

Sponsor: Zoom Sales Team International

Developer: Zoom Video Communications INC.

ST Version: 1.8

Date: 15-12-2021

CC Version: 3.1 Revision 5

Assurance Level: EAL 2

Certification ID: BSI-DSZ-CC-1173

TOE REFERENCE

TOE Name: Zoom Application

Table 1: TOE-Versions of the different clients

Client	Version
Windows-Client (64-Bit)	5.6.6
macOS-Client (Intel architecture)	5.6.6
Android-Client	5.6.6
iOS/iPadOS-Client	5.6.6

TOE Overview

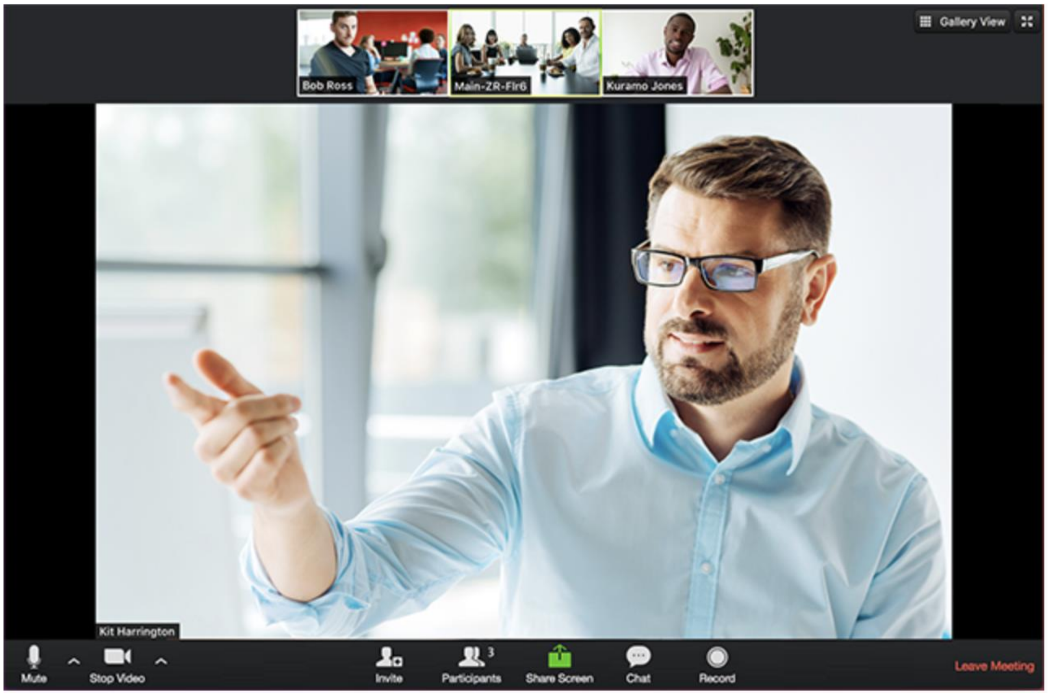
The TOE is a multi-platform software application used to host, run and organize enterprise web video communications (web meetings), with an easy, reliable cloud platform (Zoom Backend) for video and audio conferencing, collaboration, chat across mobile devices and desktops. The Zoom Backend is not part of the TOE, but of its environment.

Web meetings are primarily used to offer audio and video conferencing as well as desktop sharing. Additionally, during web meetings a user can share files and text messages with the other participants.

To access a web meeting, participants must know a randomly generated Meeting-ID as well as the meeting password set by the meeting's host. All communication between the TOE and the Zoom backend is protected by either TLS or Zoom's meeting encryption. For meeting encryption, the traffic of all meeting participants is encrypted using a symmetric meeting key generated by the Zoom backend. During a meeting, the meeting host, and if present the meeting co-host have access to certain security controls, e.g. admit participant to the meeting room, muting all participants but themselves, allowing or prohibiting screen sharing and recordings. Those controls are set by the respective users using the TOE but can only be enforced on server-side. Meeting participants can also access controls of their own, like muting themselves, starting screen share or enabling their video stream. The controls and their associated actions are protected by a role-based access control mechanism enforced by the TOE.

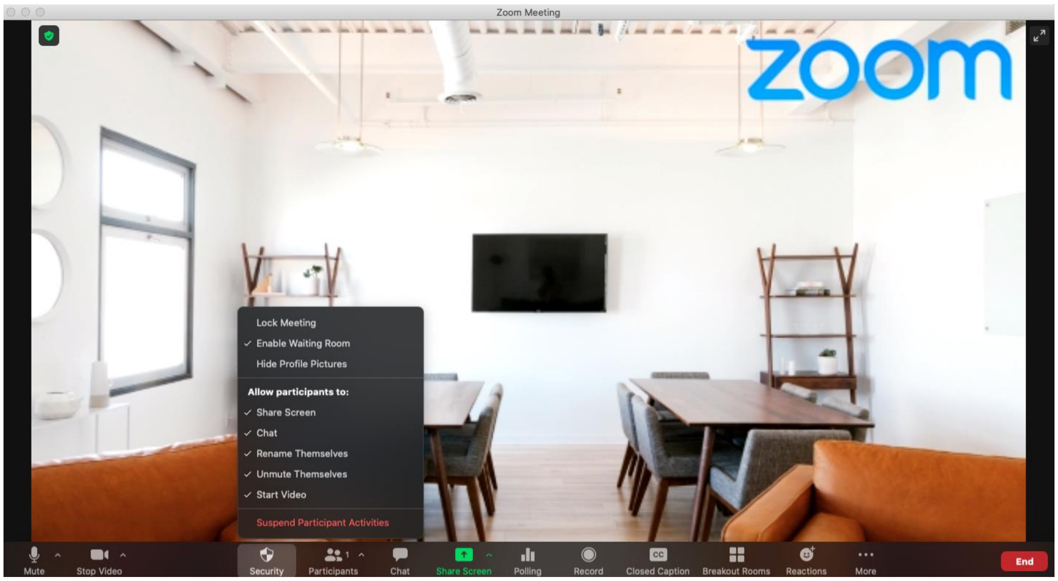
Figure 1 below shows the default view of a meeting participant:

Figure 1: Default participant view



For a meeting host, there is more functionality available as shown in Figure 2:

Figure 2: Host view during a meeting



As can be seen in Figure 2, the TOE provides the meeting host with additional settings under the tab "Security". Those security settings provide the meeting host and co-host the possibility to

- Lock meeting
- Enable waiting rooms
- Allow participants to:
  - Share screen
  - Chat
  - Rename themselves
  - Unmute themselves
  - Start video

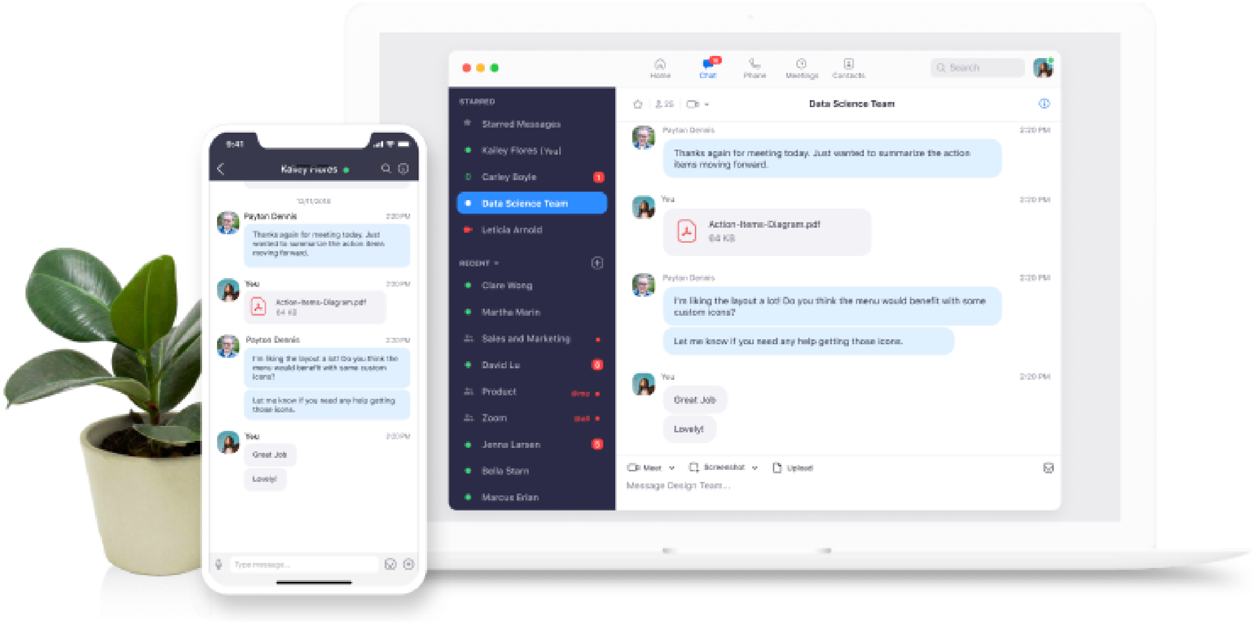
The TOE also offers an out-of-meeting instant messaging service called Zoom Chat, which can be used to share text, audio and video messages with one or more TOE users outside of web meetings. It is also possible share files with other users outside of web meetings by uploading them to Zoom Chat.

The Zoom Chat requires the TOE to keep contact and channel lists as part of the user data. The data is stored on the TOE's host device, an additional copy is also stored on the Zoom backend. As described before, the transport of the data from Zoom backend to the TOE is secured by TLS. Additionally, the TOE requires user authentication by username and password (with enforced password limitations) and also offers use of Multi-factor authentication. Messaging data (text and audio messages) is stored on the TOE's host device in an encrypted database. Additionally, for backup reasons, each encrypted message that is sent or received by the TOE is also stored on the Zoom backend using AWS Server Side Encryption (SSE)[1].

Figure 3 shows the default view for Zoom Chat:

[1] https://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html

Figure 3: Default view for Zoom Chat



Paytan Dennis 2:20 PM
Thanks again for meeting today. Just wanted to summarize the action items moving forward.

You 2:20 PM
Action-Items-Diagram.pdf
64 KB

Paytan Dennis 2:20 PM
I'm liking the layout a lot! Do you think the menu would benefit with some custom icons?

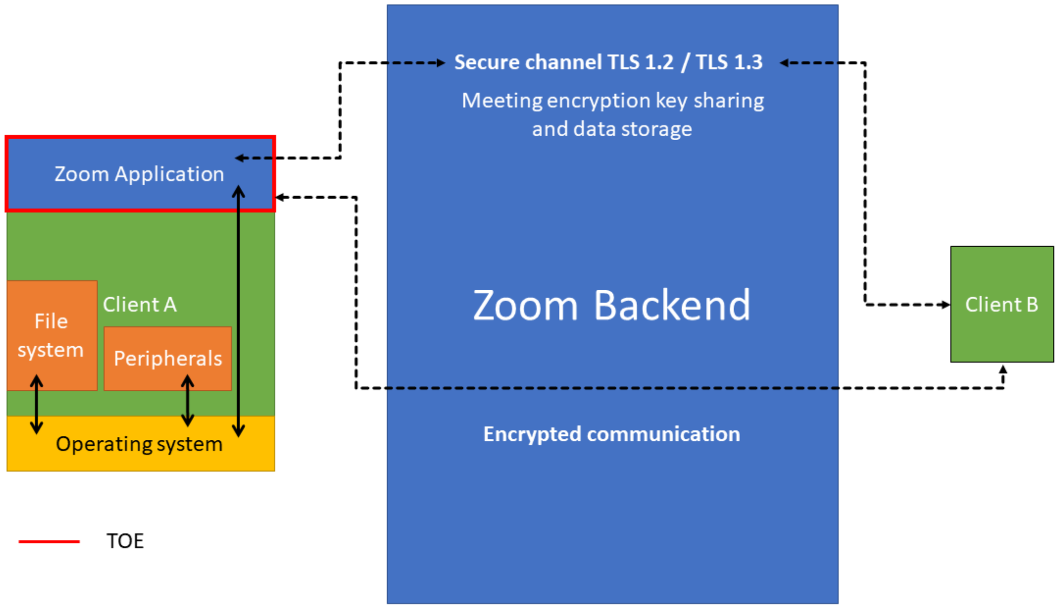
Let me know if you need any help getting those icons.

You 2:20 PM
Great Job

Lovely!

The diagram in Figure 4 provides a high-level overview of the main components acting in a Zoom meeting communication.

Figure 4: Overview of main components



Note that “Client B” is not described as detailed as “Client A” to simplify the image. They are both TOE instances.

The TOEs main security features are as follows:

- Secure user authentication,
- Protection of confidentiality and integrity of all data transferred during Zoom Meetings,
- Enforcing access control rules during Zoom Meetings, e.g. prohibiting users from unmuting their microphone,
- Enforcing of user controls, e.g. making sure no audio data is transferred if a user mutes his microphone during a Zoom Meeting,
- Protection of integrity and confidentiality of all data exchanged during Zoom Chats, including protection from Zoom itself by offering optional end-to-end encryption for Zoom Chats,
- Protection of Zoom Chat data (e.g. chat messages, voice notes) stored on the user’s device.

The TOE can be run on the following platforms:

Table 2: Supported Environments

Operating System	Supported Environment
Microsoft Windows	Windows 10 Home, 10 Pro and 10 Enterprise
Apple macOS	macOS 10.13 (Sierra), 10.14 (Mojave), 10.15 (Catalina), 11 (Big Sur)
Apple iOS/iPadOS	iOS 12, 13 and 14, iPadOS 13 and 14
Android	Android 8.1, 9, 10 and 11

Note that a Linux version of the TOE is also available. However, this version is not in scope for the certification at this time and will therefore not be discussed in the following sections.

Licensing Information

The Zoom Application can be operated with a free and a paid licence. The certified version (the TOE) needs to be operated with a paid license. The binary of the Zoom application is the same for both licensing models, the only technical difference in terms of security is, that users with a free license cannot initiate end-to-end encrypted Zoom Chats, they can however still participate in end-to-end encrypted chats if a user with a paid license initiated the chat.

All threats addressed by the TOE can also be mitigated with the free licensing, paid licencing only offers an additional layer of security for Zoom Chat.

Required Non-TOE hardware and software

The TOE needs to run on either a computer running Windows or macOS or a mobile device running Android or iOS/iPadOS. To perform cryptographic operations the TOE makes use of OpenSSL version 1.1.

All devices the Zoom application is installed on need to be part of a managed client infrastructure that controls which applications can be installed on the device

The TOE also requires the Zoom backend to operate.

For Android devices it is required that

1) The TOE is run in a managed environment, i.e:
   a) the device supports an enterprise container technology (e.g. Android for Work[2], Samsung Knox) and the TOE is only run in a managed work profile, or
   b) the TOE is run on an exclusive, managed enterprise device,
2) the device makes use of an embedded security module

The following devices (Table 3) are in scope for this certification. The devices are taken from the list of devices that Google labelled as “android enterprise recommended”[3]. Devices on this list need to provide regular security updates to the user. Also, being on that list grants a higher level of assurance that the used Android version has not been altered by the vendor compared to Google’s version.

Table 3: Supported Android mobile devices

Vendor	Model	Secure Element
Xiaomi	Xiaomi Mi 10T Pro	Qualcomm Secure Processing Unit
Google	Google Pixel 4XL	Titan M Chip

TOE DESCRIPTION

Physical Scope

The TOE is a software application expected to run on a Windows or macOS computer or an Android or iOS device. The TOE as a software application does not have any physical boundaries. The complete TOE in terms of the Common Criteria includes:

- the software application, in form of an executable program, delivered as described in Table 4,
- an operational manual, available as a download from Zoom’s website[4] in form of a PDF file [AGD].

Since there are four different platforms, there are also four TOE configurations. Each configuration consists of the TOE installed on the respective platform.

Table 4: Delivery overview for the different platforms

Platform	Way of delivery
Microsoft Windows [TOE-Windows]	Download from Zoom’s website
Apple macOS [TOE-Mac]	Download from Zoom’s website
Android [TOE-Android]	Download from Zoom’s website or Google Play Store
Apple iOS/iPadOS [TOE-ios]	iOS App Store

[2] https://www.android.com/enterprise/

[3] https://androidenterprisepartners.withgoogle.com/devices/#!?device_categories=knowledge_worker&region_names=europe

[4] https://zoom.us/

Logical Scope

The TOE only includes the Zoom Meeting and Zoom Chat functionality of the Zoom application. The Zoom backend is explicitly not part of the TOE.

Admin management is not part of the Zoom application and therefore not TOE functionality.

The enforcement of security controls is done by the Zoom application. The enforcement settings are set by the meeting host (in the Zoom application) and then stored server-side. The security controls to enforce are received from server-side.

Random number generation for cryptographic operations is left to the TOE's environment.

Zoom Meetings

Zoom Meetings are the main feature of the TOE. They are primarily used to offer audio and video conferencing as well as desktop sharing. Additionally, during web meetings a user can share files and text messages with the other participants.

To access a meeting, users authenticate themselves to the Zoom backend and then the TOE connects to a Multimedia Router (MMR). The MMR is the TOE's main communication partner during meetings and assigns each user one of three roles: Host, Co-Host or Participant.

The Host is usually the user who scheduled the meeting, while Co-Hosts are participants that the Host wants to give Host-Controls to. Host-Controls include the ability to decide which users are able to talk, show their camera or screen and record the meeting. Additionally, Host and Co-Hosts have access to the "Security Option" of a meeting (see Figure 2). Please note, that the functionalities

- Lock Meeting,
- Enable Waiting Room,
- Hide Profile Pictures,
- Chat and
- Rename themselves

are not security features in the sense of this certification, but out-of-scope features of the Zoom Client.

Zoom Chat

Zoom Chat is an out-of-meeting instant messaging service, which can be used to share text, audio and video messages with one or more TOE users outside of web meetings.

All Chats are by default TLS encrypted. The TOE also stores chat messages in a local database that is encrypted and integrity protected. Additionally, user can enable Advanced Chat Encryption (ACE) which provides end-to-end encrypted chats. If activated, the TOE performs a key agreement with the receiving user and encrypts the message making use of a shared secret. Chats using ACE are still TLS protected during transfer. Zoom Chat uses the XMPP protocol to transmit messages between users through the Zoom backend.

Security Features

- **Authentication**

  The TOE supports password-based authentication with the option to enable Two-Factor authentication using time-based one-time passwords.

- **Protection of confidentiality and integrity during Zoom Meetings**

  All meeting data transferred during Zoom Meetings is encrypted using AES256-GCM using a symmetric key generated and distributed by the Zoom backend. All other communication between the TOE and Zoom backend is protected by TLS 1.2 or TLS 1.3.

- **Enforcing of access control rules during Zoom Meetings**

  The TOE maintains the current meeting security attributes, i.e. a list of permissions the user has. The meeting security attributes can be manipulated by the Host and Co-Host of a meeting. Changes of the meeting security attributes are sent by the Host (or Co-Host) to the MMR in the Zoom Backend. From there the updated list of permissions is distributed to each user. For each user action covered by the meeting security attributes, the TOE checks if the user is permitted to perform said action. If not, the action is cancelled by the TOE. Additionally, UI elements are greyed out if the user is not permitted to perform the associated action.

- **Enforcing of user controls during a Zoom Meeting**

  Similarly to the methods described above, the TOE also keeps a list of the user's settings (User Controls) during a meeting. Every time the TOE captures input data from e.g. the microphone, it checks, if the user has their microphone enabled or disabled. If the microphone is disabled, the input data is dismissed and not sent to the MMR. The same applies for video and screen share.

- **Protection of integrity and confidentiality during Zoom Chats**

  Messages exchanged between users using Zoom Chat are always protected by TLS. Additionally, users can choose to make use of Advanced Chat Encryption which enables end-to-end encryption in Zoom Chats. In this case the users participating in the Zoom Chat perform a Diffie-Hellman key agreement using long-term elliptic curve keys created during TOE installation to derive a shared secret key. The necessary public keys are stored in the Zoom backend. This key is then used to encrypt messages using AES256-CBC encryption.

- **Protection of Zoom Chat data stored on the user's device**

  Every message sent and received through Zoom Chats is stored in a local database. The database is encrypted using ASE256-CBC and integrity protected using HMAC-SHA512. The keys used for encryption and integrity protection are securely stored using OS functionality.

CONFORMANCE CLAIMS

COMMON CRITERIA CONFORMANCE CLAIM

This Security Target claims conformance to:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2017-04-001, Version 3.1, Revision 5, April 2017 [CC1]
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2017-04-002, Version 3.1, Revision 5, April 2017 [CC2]
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; CCMB-2017-04-003, Version 3.1, Revision 5, April 2017 [CC3]

as follows:

- CC Part 2 extended
- CC Part 3 conformant (EAL 2)

PROTECTION PROFILE CONFORMANCE CLAIM

This Security Target does not claim conformance to any Protection Profile.

CONFORMANCE RATIONALE

This ST claims conformance to CC Part 2 extended because it uses the extended component “Cryptographic key import (FCS_CKM_EXT.1)”. All other SFRs are from CC Part 2 [CC2].

The ST also claims conformance to CC Part 3 as all SARs are from the EAL package EAL2 defined in CC Part 3 [CC3].

Security Problem Definition

Assets

Table 5: Assets

ID	Description
SA.User_Credentials	The credentials a TOE user uses to access his account, i.e. username and password.
SA.User_Data	A TOE user's associated data, e.g. e-mail, name, license information, contact lists, billing information and user profile info (i.e. first name, last name, profile picture, department, language preferences, IP address, location).
SA.Meeting_Credentials	The credentials a TOE user uses to access a meeting. These are the Meeting-ID and password.
SA.Meeting_Data	The data transferred between a TOE user and the Zoom backend during a meeting, e.g. voice streams, video/camera streams, shared screen, chats, file shared during a meeting.
SA.Meeting_Meta_Data	The meta data transferred between a TOE user and the Zoom backend during a meeting used for troubleshooting and statistics purposes such as active users, meetings duration, network type (e.g. Wi-Fi, LAN). Zoom Quality-of-Service (QoS) reports for past Zoom meetings collect the following information: device type, OS type and version, CPU memory, graphics card, device display info, network type, microphone, speaker selected, camera selected.
SA.Meeting_Key	The key used to encrypt communication during a meeting.
SA.Meeting_Data_To_Backend	Meeting data that is to be stored in the backend, e.g. chat messages, shared files, meeting recordings.
SA.User_Controls	The meeting controls chosen by a meeting participant, e.g. audio mute, video/screen shared.
SA.Zoom_Chat_Data	The data transferred between a TOE user and the Zoom backend during Zoom Chat such as text messages, voice and video recordings.
SA.Stored_Zoom_Chat_Data	Zoom Chat data stored on TOE's host device. This data includes text messages as well as video and voice recordings.

Threats

Table 6: Threats

ID	Description
T.Access_Login	An attacker bypasses the security functions of the TOE to gain unauthorized access to SA.User_Data and SA.User_Credentials.
T.Access_Meeting	An attacker gains access to a Zoom meeting that he is not authorized to join, therefore gaining access to SA.Meeting_Data, SA.Meeting_Credentials or SA.Meeting_Key.
T.Network	An attacker hijacks the communication between the Zoom backend and the TOE and manipulates or gains access to SA.Meeting_Data, SA.Meeting_Data_To_Backend, SA.Meeting_Meta_Data, SA.Meeting_Key, SA.Zoom_Chat_Data.
T.Bypass_User_Controls	A meeting participant gains access to shared audio, camera or screen share data, even though SA.User_Controls is set to not share that data.
T.Access_Stored_Data	An attacker gains access to the TOE's host device and accesses SA.Stored_Zoom_Chat_Data.

Organizational Security Policies

Table 7: Organizational Security Policies

ID	Description
OSP.Meeting_Password_Policy	Meeting Hosts shall not choose weaker meeting passwords than the randomly generated one (initial password).

Assumptions

Table 8: Assumptions

ID	Description
A.Meeting_Key	The Zoom backend generates SA.Meeting_Key and provides at least 120 bit security.
A.User_Credentials	Any user of the TOE does not disclose their authentication or meeting credentials to any individual not authorized for access to the TOE or meeting.
A.Rate_Limiting	The Zoom backend implements rate limiting on brute-force-attacks for meeting password and user authentication.
A.Secure_Backend	The Zoom backend is trusted. Data sent from the Zoom backend is assumed to be with integrity.
A.Host_Device	The TOE's host device of the TOE offers means to securely store and access cryptographic keys, as well as

	provide a random number generator of appropriate strength to be used for cryptographic operations required by the TOE.
A.Managed_Device	The TOE runs on a managed device. The device management controls which applications can be installed on the device, to ensure no malicious applications are installed which could harm the security of the TOE. For Android smartphones this means, the device has to support an enterprise container technology, and the TOE runs in a work profile or on an exclusive, managed enterprise device.
A.Proper_User	The user of the application is not wilfully negligent or hostile and uses the TOE within compliance of the applied enterprise security policy.
A.Non_Hostile_Platform	The platform on which the TOE is running on does not start any attacks on assets protected by the TOE.

Security Objectives

Security Objectives for the TOE

Table 9: Security Objectives for the TOE

ID	Description
O.Secure_Channel	The connection between the TOE and the Zoom backend shall provide a secure trusted channel so that confidentiality and integrity of all information transmitted when the user accesses through the TOE are protected.
O.Meeting_Encryption	The TOE shall provide strong encryption for all data transmitted during a meeting using SA.Meeting_Key.
O.Secure_Storage	The TOE shall ensure Zoom Chat data stored on the TOE's host device is never stored in plain text.
O.Meeting_Authentication	The TOE shall ensure only authorized users can access a meeting.
O.Authentication	The TOE shall ensure only authenticated users can access user data.
O.User_Control	The TOE shall enforce the SA.User_Controls set for each meeting participant.
O.User_Control_Remote	The TOE shall implement access control, so users can only use the controls according to the meeting security attributes.
O.Advanced_Chat_Encryption	The TOE shall implement end-to-end encrypted chats between members of the same organization.

Security Objectives for the TOE-environment

Table 10: Security Objectives for the TOE-environment

ID	Description
OE.Meeting_Password_Policy	Users of the TOE shall not weaken the automatically generated password when creating a meeting.
OE.Keys	The Zoom backend shall generate SA.Meeting_Key and provide at least 120 bit security.
OE.User_Credentials	Users of the TOE shall not disclose their authentication or meeting credentials to any individual not authorized for access to the TOE or meeting.
OE.Rate_Limiting	The Zoom backend shall implement rate limiting after a certain amount of failed authentication attempts for either user or meeting authentication.
OE.Secure_Key_Storage	The host device of the TOE shall provide means to securely store and access cryptographic keys.
OE.Meeting_Security_Attributes	The Zoom backend shall store, protect the integrity and update meeting security attributes for authenticated meeting hosts. Meeting security attributes include

ID	Description
	permissions of participants (e.g. who is allowed to share their screen) as well as the user identification who is meeting host.
OE.Meeting_Security_Controls	The Zoom backend shall enforce the security controls set by meeting Host and Co-Host.
OE.Password_Strength	The Zoom backend shall ensure that user selected passwords are complex.
OE.Managed_Device	The organization using the TOE shall make sure the TOE is only installed on managed devices. The client management shall ensure no malicious applications are installed on the device.
OE.Proper_User	The user of the application shall not be wilfully negligent or hostile and use the TOE within compliance of the applied enterprise security policy.
OE.Secure_Server_Storage	The Zoom backend shall protect the confidentiality and integrity of all data sent to be stored in it.
OE.RNG	The host device shall provide a random number generator of appropriate strength to be used for cryptographic purposes required by the TOE. Note: Due to the multitude of supported platforms, users of the TOE are able to choose which platform (including its RNG) they use.

Security Objectives Rationale

Table 11: Security Objectives Rationale

	O.Secure_Channel	O.Meeting_Encryption	O.Secure_Storage	O.Meeting_Authentication	O.Authentication	O.User_Control	O.User_Control_Remote	O.Advanced_Chat_Encryption	OE.Meeting_Password_Policy	OE.Keys	OE.User_Credentials	OE.Rate_Limiting	OE.Secure_Key_Storage	OE.Meeting_Security_Attributes	OE.Meeting_Security_Controls	OE.Password_Strength	OE.Managed_Device	OE.Proper_User	OE.Secure_Server_Storage	OE.RNG
T.Access_Meeting	x			x					x		x	x								x
T.Access_Login	x				x						x	x				x				x
T.Network	x	x						x		x							x			x
T.Bypass_User_Controls						x	x							x						
T.Access_Stored_Data			x										x				x			x
OSP.Meeting_Password_Policy									x											
A.Meeting_Key										x										
A.User_Credentials											x									
A.Rate_Limiting												x								
A.Secure_Backend														x	x				x	
A.Host_Device													x							x
A.Managed_Device																	x			
A.Proper_User																		x		
A.Non_Hostile_Platform																	x			

T.Access_Meeting

O.Meeting_Authentication ensures meetings can only be accessed by entering the correct Meeting-ID and Meeting-Password. **O.Secure_Channel** prevents attackers from gaining access to that information by listening into the communication. The Secure Channel is supported by **OE.RNG** for generation of random numbers. **OE.User_Credentials** prevents users from sharing meeting credentials with any user not meant to take part in the meeting. **OE.Rate_Limiting** hinders the performance of a brute-force-attack on the meeting password. The attack vector is further weakened by **OE.Meeting_Password_Policy** which prevents use of weak meeting passwords.

T.Access_Login

O.Authentication enforces authentication of users by at least username and password. Optionally authentication can be strengthened by use of multi-factor-authentication. **OE.Password_Strength** requires the used password to be cryptographically strong to prevent password guessing attacks. **O.Secure_Channel** ensure passwords are only transferred to the Zoom backend in encrypted form, to prevent potential Man-in-the-Middle attacks. The Secure Channel is supported by **OE.RNG** for generation of random numbers. **OE.User_Credentials** prevents users from sharing their user credentials with any other person. **OE.Rate_Limiting** weakens brute-force attacks.

T.Network

O.Secure_Channel requires all communication between TOE and Zoom backend to be encrypted and integrity-protected. During meetings an additional layer of encryption is used (**O.Meeting_Encryption**). That encryption is possible due to the secure server-generated key (**OE.Keys**). Zoom Chat is encrypted using the **O.Secure_Channel**. As an additional layer of protection Advanced Chat Encryption can be enabled (**O.Advanced_Chat_Encryption**). The key generation for Advanced Chat Encryption and the secure channel is supported by **OE.RNG**. **OE.Managed_Device** ensures only trusted software is installed on the TOE's host device that won't try to access data in the communication channel.

T.Bypass_User_Controls

O.User_Control_Remote enforces access controls during a meeting according to the meeting security attributes managed by **OE.Meeting_Security_Attributes**. **O.User_Control** enforces the controls set by the participant.

T.Access_Stored_Data

O.Secure_Storage makes sure data is only stored on the TOE's host device in encrypted form. The key used to encrypt the data is generated with the help of **OE.RNG**. Key storage is done with **OE.Secure_Key_Storage**. **OE.Managed_Device** ensures only trusted software is installed on the TOE's host device that won't try to access data stored on the device.

OSP.Meeting_Password_Policy

OE.Meeting_Password_Policy makes sure users chose meeting passwords that are not weaker compared to the automatically generated (initial) meeting passwords.

A.Meeting_Key

A.Meeting_Key is met by the equally named **OE.Keys**.

A.User_Credentials

A.User_Credentials is met by the equally named **OE.User_Credentials**.

A.Rate_Limiting

A.Rate_Limiting is met by the equally named **OE.Rate_Limiting**.

A.Secure_Backend

A.Secure_Backend requires the Zoom backend to be trusted and to protect the data stored on it. This is achieved by **OE.Secure_Server_Storage**. Additionally, both **OE.Meeting_Security_Attributes** and **OE.Meeting_Security_Controls** protect the meeting security attributes.

A.Host_Device

A.Host_Device requires the TOE's host device to provide means to securely store and access cryptographic keys. This is achieved by **OE.Secure_Key_Storage**. It also requires the host device to provide a random number generator of appropriate strength to be used for cryptographic operations. This is fulfilled by **OE.RNG**.

A.Managed_Device

A.Managed_Device is met by the equally named **OE.Managed_Device**.

A.Proper_User

A.Proper_User is met by the equally named **OE.Proper_User.**

A.Non_Hostile_Platform

By providing a managed runtime environment for the TOE, **OE.Managed_Device** also ensures the platform is not hostile as required by **A.Non_Hostile_Platform**.

EXTENDED COMPONENTS DEFINITION

CRYPTOGRAPHIC KEY IMPORT (FCS_CKM_EXT.1)

Family Behaviour

The family FCS_CKM_EXT is an enhancement of FCS_CKM defined in CC Part 2 [CC2]. It describes rules for key import. Key import is the process by which one or more keys are generated on a trusted, non-TOE platform and imported to the TOE over a trusted channel.

Component levelling

The family FCS_CKM_EXT.1 only contains one component FCS_CKM_EXT.1.

Management FCS_CKM_EXT.1

There are no management activities foreseen.

Audit FCS_CKM_EXT.1

There are no audit activities foreseen.

FCS_CKM_EXT.1 Cryptographic key import

Hierarchical to:	No other components
Dependencies:	FCS_CKM.4 Cryptographic key destruction FCS_COP.1 Cryptographic operation [FTP_ITC.1 Inter-TSF trusted channel or FTP_TRP.1 Trusted path]

FCS_CKM_EXT.1.1 The TSF shall import cryptographic keys from another trusted IT-entity which can be used for [assignment: cryptographic operation].

SECURITY REQUIREMENTS

POLICIES

User Control policy

The User Control Policy is defined as:

1 Purpose: To control access to user controls during a meeting

2 Subjects: Participants of a meeting

3 Information: User controls (SA.User_Controls)

4 Security attributes: State of the control, i.e. enabled or disabled

5 SFR instances: FDP_ACC.1/UC, FDP_ACF.1/UC

SFRS

Identification and authentication

FIA_UID.1 Timing of identification

Hierarchical to: No other components

Dependencies: No dependencies.

FIA_UID.1.1 The TSF shall allow *accessing and changing general settings, including setting the defaults for activated camera and audio during* meetings[5] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.1/Client Timing of identification

Hierarchical to: No other components

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.1.1/Client The TSF shall allow *accessing and changing general settings, including setting the defaults for activated camera and audio during* meetings[6] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2/Client The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

ST application note 1: This SFR applies to user authentication to the TOE

[5] [assignment: list of TSF-mediated actions]

[6] [assignment: list of TSF-mediated actions]

FIA_UAU.2/Meeting **User authentication before any action**

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.2.1/Meeting The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

ST application note 2: This SFR applies to user authentication to a meeting. To authenticate a user needs both the Meeting-ID and the password set by the meeting Host. In addition to entering this data in the Zoom Client users can join meetings by using a meeting invite link or by accepting a meeting invitation in the Zoom client. Both ways contain the Meeting ID and password in an encoded format.

FIA_UAU.5 **Multiple authentication mechanisms**

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.5.1 The TSF shall provide *password-based authentication, Time-based one-time passwords*[7] to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the *following rules:*

1) Password-based authentication,
2) *Authentication with time-based one-time passwords can be enabled by TOE users as a second factor in case of password-based authentication*.[8]

Access Control

FDP_ACC.1/UC **Subset access control**

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/UC The TSF shall enforce the *User Control Policy*[9] on

1 subjects: Participants;
2 objects: SA.User_Controls*;*
3 operations: Enabling/Disabling of user controls[10]*.*

FDP_ACF.1/UC **Security attribute based access control**

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

[7] [assignment: list of multiple authentication mechanisms]

[8] [assignment: rules describing how the multiple authentication mechanisms provide authentication]

[9] [assignment: access control SFP]

[10] [assignment: list of subjects, objects, and operations among subjects and objects covered by the SFP]

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/UC The TSF shall enforce the *User Control Policy*[11] to objects based on the following:

1. *subjects: Participants;*
2. *objects:* SA.User_Controls[12].

FDP_ACF.1.2/UC The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *The right to enable or disable user controls is given according to the meeting security attributes*[13].

FDP_ACF.1.3/UC The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *The Host of a meeting has full access and can enable/disable his own user controls without limitations*[14].

FDP_ACF.1.4/UC The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *none*[15].

FMT_MSA.3/UC **Static attribute initialisation**

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MSA.3.1/UC The TSF shall enforce the *User Control Policy*[16] to provide *permissive*[17] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/UC The TSF shall allow the *Host*[18] to specify alternative initial values to override the default values when an object or information is created.

ST application note 3: Initial values in this context means the initial values during a meeting. E.g. a host can mute all participants when they join, and can prevent them from unmuting themselves.

FMT_MSA.1 **Management of security attributes**

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or

[11] [assignment: access control SFP]

[12] [assignment: list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes]

[13] [assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]

[14] [assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]

[15] [assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]

[16] [assignment: access control SFP, information flow control SFP]

[17] [selection, choose one of: restrictive, permissive, [assignment: other property]]

[18] [assignment: the authorised identified roles]

FDP_IFC.1 Subset information flow control]

FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 The TSF shall enforce the *User control policy*[19] to restrict the ability to *change_default*[20] the security attributes *SA.User_Controls*[21] to *Host*[22].

FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles *Authenticated User, Non-Authenticated User, Host, Co-Host, Participant*[23].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

ST application note 4: The roles Host, Co-Host and Participant are the roles TOE users get assigned during a meeting. The roles Non-Authenticated User and Authenticated User are roles a TOE user can have in respect to the TOE

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- *Mute/Unmute microphone,*
- *Enable/Disable camera,*
- *Enable/Disable screen sharing,*
- Enable/Disable meeting recording[24].

Cryptographic operations - General

FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured

[19] [assignment: access control SFP(s), information flow control SFP(s)]

[20] [selection: change_default, query, modify, delete, [assignment: other operations]]

[21] [assignment: list of security attributes]

[22] [assignment: the authorised identified roles]

[23] [assignment: the authorised identified roles]

[24] [assignment: list of management functions to be provided by the TSF]

identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *the TSF*[25] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for *any communication between Zoom backend and Zoom application except for the transfer of meeting data*[26].

ST application note 5: The trusted channel as defined above is a TLS-connection between Zoom application and Zoom backend. The TOE must implement TLS 1.2 and TLS 1.3 as defined in [RFC 5246, RFC 8446] and shall only use the following Cipher-Suites:

Table 12: List of used TLS Cipher Suites

Protocol Version	Cipher Suite
TLS 1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
TLS 1.3	TLS_AES_256_GCM_SHA384
	TLS_CHACHA20_POLY1305_SHA256
	TLS_AES_128_GCM_SHA256

Additionally the client shall send the signalling cipher suite TLS_EMPTY_RENEGOTIATION_INFO_SCSV to protect itself from renegotiation attacks

Mutual authentication is done by the following mechanisms: The client authenticates the server through server certificates, the server authenticates the client through user authentication (FIA_UAU.1/Client).

ST application note 6: The TOE makes use of long-lived sessions in which the Zoom backend may initiate communication to the TOE but the session has to be started by the TOE.

ST application note 7: The trusted channel is also used to transmit SA.Meeting_Meta_Data. The TOE shall only collect and transmit the following data:

- Topic,
- Description (Optional),
- Time,
- Date,
- Duration,
- Time Zone,

[25] [selection: the TSF, another trusted IT product]

[26] [assignment: list of functions for which a trusted channel is required]

- Participant IP Addresses,
- Device/Hardware Information,
- Meeting Statistics/Metrics.
- Start Time,
- Join Time,
- Leave Time

Cryptographic operations – Zoom Meetings

FCS_CKM_EXT.1 **Cryptographic key import**

Hierarchical to: No other components

Dependencies: FCS_CKM.4 Cryptographic key destruction

FCS_COP.1 Cryptographic operation

[FTP_ITC.1 Inter-TSF trusted channel or

FTP_TRP.1 Trusted path]

FCS_CKM_EXT.1.1 The TSF shall import cryptographic keys from another trusted IT-entity which can be used for *AES-256 encryption*[27].

FCS_COP.1/AES-GCM **Cryptographic operation**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/AES-GCM The TSF shall perform *symmetric encryption*[28] in accordance with a specified cryptographic algorithm *AES-GCM*[29] and cryptographic key sizes 256 bit[30] that meet the following: *NIST FIPS 197 [NIST FIPS 197], NIST SP 800-38D [NIST SP 800-38D]*[31].

ST application note 8: This Iteration applies to enhanced encryption during a meeting.

Cryptographic operations – Zoom Chat

[27] [assignment: cryptographic operation]

[28] [assignment: list of cryptographic operations]

[29] [assignment: cryptographic algorithm]

[30] [assignment: cryptographic key sizes]

[31] [assignment: list of standards]

FCS_CKM.1/ACE **Cryptographic key generation**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or

FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/ACE The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *ECC key generation with curve P-521*[32] and specified cryptographic key sizes *521 Bit*[33] that meet the following: *FIPS PUB 186-4 B.4 and D.1.2.5 [NIST FIPS 186-4]*[34].

ST application note 9: This is the asymmetric key pair used for Advanced Chat Encryption.

FCS_CKM.1/Derivation **Cryptographic key generation**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or

FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1/Derivation The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *Elliptic Curve Diffie Hellman with SHA-256*[35] and specified cryptographic key sizes *256 Bit*[36] that meet the following: *NIST SP 800-56A Rev. 3 [NIST SP 800-56A3], NIST FIPS 180-4 [NIST FIPS 180-4]*[37].

ST application note 10: These are session keys used to transmit symmetric keys in ACE.

FCS_CKM.1/AES **Cryptographic key generation**

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or

FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

[32] [assignment: cryptographic key generation algorithm]

[33] [assignment: cryptographic key sizes]

[34] [assignment: list of standards]

[35] [assignment: cryptographic key generation algorithm]

[36] [assignment: cryptographic key sizes]

[37] [assignment: list of standards]

FCS_CKM.1.1/AES The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *symmetric key generation*[38] and specified cryptographic key sizes *256 Bit*[39] that meet the following: *NIST FIPS 197 [NIST FIPS 197]*[40].

ST application note 11: AES keys are used on multiple occasions. It is used to encrypt data during Zoom Chats using Advanced Chat Encryption as well as during encryption of the local database.

FCS_COP.1/AES-CBC **Cryptographic operation**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1/AES-CBC The TSF shall perform *symmetric encryption*[41] in accordance with a specified cryptographic algorithm *AES-CBC*[42] and cryptographic key sizes *256 Bit*[43] that meet the following: *NIST FIPS 197 [NIST FIPS 197], NIST SP 800-38A [NIST SP 800-38A]*[44].

ST application note 12: This iteration applies to encryption of local database as well as encryption during of messages during Advanced Chat Encryption.

FCS_COP.1/HMAC **Cryptographic operation**

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or

FDP_ITC.2 Import of user data with security attributes, or

FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

[38] [assignment: cryptographic key generation algorithm]

[39] [assignment: cryptographic key sizes]

[40] [assignment: list of standards]

[41] [assignment: list of cryptographic operations]

[42] [assignment: cryptographic algorithm]

[43] [assignment: cryptographic key sizes]

[44] [assignment: list of standards]

FCS_COP.1.1/HMAC The TSF shall perform *HMAC generation and verification*[45] in accordance with a specified cryptographic algorithm *HMAC-SHA512*[46] and cryptographic key sizes *256 Bit*[47] that meet the following: *RFC 2104 [RFC2104], NIST FIPS 180-4 [NIST FIPS 180-4]*[48].

ST application note 13: HMAC shall be used for integrity protection of the encrypted local database (FCS_COP.1/AES-CBC).

SARs

This ST requires the TOE to be evaluated according to EAL2.

Security Requirements Rationale

Dependency Rationale

Table 13: Dependency Rationale

SFR	Dependencies	Satisfied by	Explanation
FIA_UID.1	No dependencies	-	-
FIA_UAU.1/Client	FIA_UID.1	FIA_UID.1	-
FIA_UAU.2/Meeting	FIA_UID.1	FIA_UID.1	-
FIA_UAU.5	No dependencies	-	-
FCS_CKM_EXT.1	FCS_CKM.4 FCS_COP.1 [FTP_ITC.1, or FTP_TRP.1]	FCS_COP.1/AES-GCM FTP_ITC.1	FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FCS_CKM.1/ACE	[FCS_CKM.2, or FCS_COP.1] FCS_CKM.4	-	The dependency to FCS_CKM.2 or FCS_COP.1 is not fulfilled because the key is only used to derive a symmetric key according to FCS_CKM.1/Derivation. FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FCS_CKM.1/Derivation	[FCS_CKM.2, or FCS_COP.1] FCS_CKM.4	FCS_COP.1/AES-CBC	FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.

[45] [assignment: list of cryptographic operations]

[46] [assignment: cryptographic algorithm]

[47] [assignment: cryptographic key sizes]

[48] [assignment: list of standards]

SFR	Dependencies	Satisfied by	Explanation
FCS_CKM.1/AES	[FCS_CKM.2, or FCS_COP.1] FCS_CKM.4	FCS_COP.1/AES-CBC	FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FCS_COP.1/AES-GCM	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM_EXT.1	FCS_CKM_EXT.1 was chosen instead of FDP_ITC.1, FDP_ITC.2 and FCS_CKM.1 because the key is imported from a trusted IT product in the TOEs environment under control of the TOEs developer. FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FCS_COP.1/AES-CBC	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/AES FCS_CKM.1/Derivation	FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FCS_COP.1/HMAC	[FDP_ITC.1, or FDP_ITC.2, or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1/AES	FCS_CKM.4 is not fulfilled because key deletion is left to the OS and is not part of TOE functionality.
FTP_ITC.1	No dependencies	-	-
FDP_ACC.1/UC	FDP_ACF.1	FDP_ACF.1/UC	-
FDP_ACF.1/UC	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/UC FMT_MSA.3/UC	-
FMT_MSA.3/UC	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 FMT_SMR.1	-
FMT_MSA.1	[FDP_ACC.1, or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/UC FMT_SMR.1 FMT_SMF.1	-
FMT_SMR.1	FIA_UID.1	FIA_UID.1	FIA_UID.1 is hierarchical to FIA_UID.1
FMT_SMF.1	No dependencies	-	-

Security functional requirements rationale

Table 14: Security functional requirements rationale

	O.Secure_Channel	O.Meeting_Encryption	O.Secure_Storage	O.Meeting_Authentication	O.Authentication	O.User_Control	O.User_Control_Remote	O.Advanced_Chat_Encryption
FIA_UID.1					x			
FIA_UAU.1/Client					x			
FIA_UAU.2/Meeting				x				
FIA_UAU.5					x			
FCS_CKM_EXT.1		x						
FCS_CKM.1/ACE								x
FCS_CKM.1/Derivation								x
FCS_CKM.1/AES		x	x					x
FCS_COP.1/AES-GCM		x						
FCS_COP.1/AES-CBC			x					x
FCS_COP.1/HMAC			x					
FTP_ITC.1	x	x						x
FDP_ACC.1/UC							x	
FDP_ACF.1/UC							x	
FMT_MSA.3/UC							x	
FMT_MSA.1							x	
FMT_SMR.1							x	
FMT_SMF.1						x		

O.Secure_Channel

The secure channel required by this security objective is provided by FTP_ITC.1.

O.Meeting_Encryption

The encryption of meeting data is done by FCS_COP.1/AES-GCM. The meeting key is imported as defined in FCS_CKM_EXT.1 over the trusted channel defined in FTP_ITC.1. That trusted channel also provides an additional layer of protection for the encrypted data.

O.Secure_Storage

Zoom Chat data gets stored in an encrypted database on the host device of the TOE. The encryption is done as defined in FCS_COP.1/AES-CBC with the key generated as in FCS_CKM.1/AES. After encryption there is also an integrity protection mechanism in form of FCS_COP.1/HMAC applied.

O.Meeting_Authentication

This objective requires the TOE to only let authenticated user join a meeting. This authentication is described in FIA_UAU.2/Meeting.

O.Authentication

O.Authentication requires the TOE to authenticate its users before accessing user data. Identification and authentication are done via FIA_UID.1 and FIA_UAU.1/Client. FIA_UAU.5 further describers the multiple authentication methods available to users.

O.User_Control

O.User_Control requires the TOE to enforce use controls set by a TOE user while taking part in a meeting. FMT_SMF.1 describes those controls and requires the TOE to implement them correctly.

O.User_Control_Remote

O.User_Control_Remote requires the TOE to enforce the access control rules to user controls according the meeting security attributes. FDP_ACC.1/UC, FDP_ACF.1/UC, FMT_MSA.3/UC and FMT_MSA.1 specify the access control rules needed. FMT_SMR.1 defines the roles used for access control.

O.Advanced_Chat_Encryption

O.Advanced_Chat_Encryption requires the TOE to implement end-to-end encryption. FCS_CKM.1/ACE creates an asymmetric key pair for the TOE user. FCS_CKM.1/Derivation derives symmetric encryption keys for chats which is used to securely transfer a session key generated by FCS_CKM.1/AES. Encryption is done as described in FCS_COP.1/AES-CBC. FTP_ITC.1 provides an additional layer of protection for the encrypted data.

Security assurance requirements rationale

The EAL2 was chosen because it provides assurance by a full security target and an analysis of the SFRs in that ST, using a functional and interface specification, guidance documentation and a basic description of the architecture of the TOE, to understand the security behaviour. The analysis is supported by independent testing of the TSF, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, and a vulnerability analysis (based upon the functional specification, TOE design, security architecture description and guidance evidence provided) demonstrating resistance to penetration attackers with a basic attack potential. EAL2 also provides assurance through use of a configuration management system and evidence of secure delivery procedures.

Together, this provides assurance commensurate with the threat environment that is experienced by typical consumers of the TOE.

TOE Summary Specification

User identification and authentication

The TOE identifies and authenticates all user before granting them access to any TSF functionality (FIA_UID.1, FIA_UAU.1/Client). Authentication is done via password-based authentication, optionally strengthened by two-factor-authentication using time-based-one-time-passwords (FIA_UAU.5).

Meeting authentication

Authentication to a meeting is password based. The user needs to be authenticated by Meeting-ID and password to be able to access any meeting functionality (FIA_UAU.2/Meeting).

User and security controls

During a meeting, users are able to access controls like muting and unmuting their microphone or starting and stopping screen sharing (FMT_SMF.1). The permissions that define what functionality is allowed to be used by each participant are stored server-side as meeting security attributes. Those can be set by the meeting's host. To enforce these permissions the TOE implements a role-based access model (FMT_SMR.1). Participants can only access functionality according to the permissions set in the meeting security attributes. The host always has full control over his settings (FDP_ACC.1/UC, FDP_ACF.1/UC). The default permissions are defined to allow every participant full access to their controls (FMT_MSA.3/UC). These default permissions can be changed by the Host on a meeting by meeting basis (FMT_MSA.1).

Secure data transfer

All data transferred between the TOE and Zoom backend is protected by either a TLS channel (FTP_ITC.1) or Zoom's meeting encryption. Key generation is supported by the RNG (OE.RNG). During Zoom's meeting encryption the TOE performs AES-GCM encryption of all meeting data (e.g. voice, video) (FCS_COP.1/AES-GCM). To perform meeting encryption, the TOE imports a key generated in the Zoom backend (FCS_CKM_EXT.1). That key is generated individually for each meeting and distributed to all participants.

Zoom Chat

Zoom Chat generally uses the same TLS channel as the rest of the TOE functionality (FTP_ITC.1). Additionally, the TOE stores Zoom Chat data in a database that is encrypted using AES-CBC and integrity protected using HMAC-SHA512 (FCS_CKM.1/AES, FCS_COP.1/AES-CBC, FCS_COP.1/HMAC).

Additionally, users can choose to make use of advanced chat encryption. For that case the TOE checks after authentication is done, if an asymmetric elliptic curve key pair already exists. If the key pair does not exist or is expired, the TOE generates a new user specific, long-term asymmetric elliptic curve key pair (FCS_CKM.1/ACE). When initiating a Zoom Chat with another user, the TOE queries the Zoom Backend for the recipient's public key, which it then uses for an elliptic curve Diffie-Hellman key exchange to derive a shared secret (FCS_CKM.1/Derivation). This shared secret is used to encrypt an ephemeral AES session key (FCS_CKM.1/AES) which can be used to secure the following communication (FCS_COP.1/AES-CBC).

USE OF CRYPTOGRAPHIC FUNCTIONS

Table 15: Cryptographic functions overview

No	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Security Level above 100 Bits
1	Meeting encryption	AES in GCM mode	FIPS 197, NIST SP 800-38D	|k| = 256	yes
2	Key Agreement	ECDH with SHA-256	NIST SP 800-56A Rev. 3, FIPS 180-4	Key sizes corresponding to the used elliptic curve P-521	yes
3	Zoom Chat encryption	AES in CBC mode	FIPS 197, NIST SP 800-38A	|k| = 256	yes
4	Database encryption	AES in CBC mode	FIPS 197, NIST SP 800-38A	|k| = 256	yes
5	Database integrity protection	HMAC-SHA512	RFC 2104, NIST FIPS 180-4	|k| = 256	yes
6	TLS 1.2 - AES 256	TLS v1.2 with the following Cipher Suites: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_DHE_RSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384, TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	RFC 5246, RFC 5288, RFC 5289	|k| = 256	yes
7	TLS 1.2 - AES 128	TLS v1.2 with the following Cipher Suites: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	RFC 5246, RFC 5288, RFC 5289	|k| = 128	yes

No	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Security Level above 100 Bits
		TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 TLS_DHE_RSA_WITH_AES_128_CBC_SHA256			
8	TLS 1.2 - ChaCha20	TLS v1.2 with the following Cipher Suites: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	RFC 5246, RFC 7905	|k| = 256	see ST application note 14:
9	TLS 1.3 - AES 256	TLS v1.3 with the following Cipher Suite: TLS_AES_256_GCM_SHA384	RFC 8446	|k| = 256	yes
10	TLS 1.3 - AES 128	TLS v1.3 with the following Cipher Suite: TLS_AES_128_GCM_SHA256	RFC 8446	|k| = 128	yes
11	TLS 1.3 - ChaCha20	TLS v1.3 with the following Cipher Suite: TLS_CHACHA20_POLY1305_SHA256	RFC 8446	|k| = 256	see ST application note 14:
12	Key generation	ECC key generation with curve P-521	FIPS PUB 186-4 B.4 and D.1.2.5	Key sizes corresponding to the used elliptic curve P-521	yes
13	Symmetric key generation	Random bit input from OE.RNG.	-	|k| = 256	yes

ST application note 14: No mathematical analysis of Cipher Suites using ChaCha20-Ploy1305 has been done.

Annex

Reference Documentation

Table 16: Reference documentation

ID	Document
AGD	Zoom Application – Guidance Documentation, Version 1.5, 06.12.2021
CC1	Common Criteria, Part 1: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, Version 3.1, Revision 5, April 2017
CC2	Common Criteria, Part 2: Common Criteria for Information Technology Security Evaluation, Part 2: Security functional components, Version 3.1, Revision 5, April 2017
CC3	Common Criteria, Part 3: Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance components, Version 3.1, Revision 5, April 2017
NIST FIPS 186-4	Federal Information Processing Standards Publication 186-4, Digital Signature Standard (DSS), July 2013
NIST FIPS 180-4	Federal Information Processing Standards Publication 186-4, Secure Hash Standard (SHS), August 2015
NIST FIPS 197	Federal Information Processing Standards Publication 197, Advanced Encryption Standard (AES), 2001
NIST SP 800-38A	NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation – Methods and Techniques, 2001
NIST SP 800-38D	NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, 2007
NIST SP 800-56A3	NIST Special Publication 800-56A Rev. 3, Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography, 2018
RFC 2104	RFC 2104 – HMAC: Keyed-Hashing for Message Authentication, https://tools.ietf.org/html/rfc2104
RFC 5246	RFC 5246 – The Transport Layer Security (TLS) Protocol Version 1.2, https://tools.ietf.org/html/rfc5246
RFC 5288	RFC 5288 - AES Galois Counter Mode (GCM) Cipher Suites for TLS, https://datatracker.ietf.org/doc/html/rfc5288
RFC 5289	RFC 5289 - TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM), https://datatracker.ietf.org/doc/html/rfc5289
RFC 7905	RFC 7905 - ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS), https://datatracker.ietf.org/doc/html/rfc7905
RFC 8446	RFC 8446 - The Transport Layer Security (TLS) Protocol Version 1.3, https://datatracker.ietf.org/doc/html/rfc8446
TOE-Windows	ZoomInstaller.exe, Version 5.6.6, SHA256: 547BB69B964DB47D84D67F006A44927CD70D2B5380C83B8D3299F0CC2B9620B5
TOE-Mac	Zoom.pkg, Version 5.6.6, SHA256: 562D4FC25FF0B3AE3DF597644CD6E2CDC0513751F610E334FF23E6464C73BF45
TOE-Android	zoom.apk, Version 5.6.6, SHA256: CD2B482CCE711EE0ED012B76F733398AC23988E457B9748494B90668E38AF02F

TOE-ios	Zoom 5.6.6.ipa, Version 5.6.6, SHA256: 64A04310D2A9E00E4C2EFC8C9E1561C681BC3B74EAD778BA0F7B59B5188AD601

TERMINOLOGY

Table 17: Terminology

Acronym	Term
ACE	Advanced Chat Encryption
CA	Certificate Authority
E2E	End-to-end
E2EE	End-to-end encryption
IM	Instant Messaging
PKI	Public Key Infrastructure
RNG	Random Number Generation
TOE	Target of Evaluation

LIST OF FIGURES

LIST OF TABLES