Information Security Auditing and Digital Forensics

Zoom Application 5.6.6

Studente: Ludovico Guercio

Studio del Security Target di un prodotto certificato Common Criteria

Prodotto scelto:

Zoom Application nella versione 5.6.6 per dispositivi Windows 64 bit, MacOS, Android e iOS-iPadOS

Descrizione del prodotto

Zoom è un software multipiattaforma utilizzato per organizzare, ospitare, eseguire web meeting (quindi video conference) appoggiandosi ad una piattaforma cloud chiamata Zoom Backend.
L'applicazione è ormai molto famosa , utilizzata e apprezzata. La principale funzionalità che si utilizza in Zoom è quella delle riunioni web: qui è possibile creare conferenze audio e video, condividere video e schermate desktop. Il prodotto inoltre fornisce la possibilità di poter condividere, tra gli utenti partecipanti ai meeting, file e messaggi di testo. Questo servizio di

messaggistica viene anche fornito anche al di fuori di un meeting, grazie a Zoom Chat.

Descrizione del TOE

Il Security Target prende in analisi l'applicazione client Zoom che è composta da due software:

- Zoom Meeting : utilizzato per organizzare video-conferenze
- Zoom Chat : utilizzato come strumento di messaggistica al di fuori dei meeting

Come anticipato precedentemente, il funzionamento dell'applicazione Zoom è permesso anche grazie a Zoom Backend. Quest'ultimo però, non fa parte del Target of Evaluation anche se è parte integrante del funzionamento dell'applicazione.

Dal TOE summary specification si può analizzare facilmente le specifiche di funzionalità che il TOE fornisce per l'uso finale.
Il TOE permette inizialmente (ad esempio al suo avvio) di eseguire la fase di autenticazione ed identificazione: questo passaggio viene descritto sia per l'autenticazione ad una riunione, sia per l'autenticazione al proprio account. Il TOE prevede che possono utilizzare l'applicazione sia utenti esterni (quindi non registrati) che utenti registrati. Per accedere ad una web-meeting i partecipanti hanno bisogno di un ID riunione (che viene generato casualmente) ed di una password per potervi accedere; anche la password è generata dal TOE stesso, ma può essere modificata dall'organizzatore. L'autenticazione al proprio account Zoom, invece, avviene semplicemente via username-password (+ autenticazione a due fattori se attivata).
Il TOE durante una riunione permette la gestione dei partecipanti e dei controlli di sicurezza. L'organizzatore e il co-organizzatore (se presente) , tramite un'interfaccia grafica ottimale, hanno accesso a determinati comandi come ad esempio l'accettare l'ingresso in riunione di un partecipante, mutare i partecipanti, consentire la condivisione schermo di un partecipante; tali controlli sono impostati tramite il TOE ma possono essere anche forzati lato server.
Dunque è presente un meccanismo di controllo degli accessi basato su ruoli che viene impostato dal TOE: ad esempio entrando in una riunione, il TOE assegna

quell'utente entrante il ruolo di "partecipante" che gli limita quindi l'uso di alcune funzionalità.
Una delle fondamentali funzionalità del TOE è quella di avere un canale sicuro di comunicazione: tutte le comunicazioni tra il TOE e il Zoom Backend sono protette da un canale TLS e crittografate: tutti i dati di un meeting vengono crittografati utilizzando una chiave della riunione simmetrica generata dal Backend.
L'ultimo servizio che offre il TOE è quello di Zoom Chat; qui oltre al semplice utilizzo di scambio di messaggi, si possono salvare contatti ed elenco di canali per conto dell'utente. I dati chat vengono salvati nel dispositivo host del TOE e viene anche effettuata una copia che verrà archiviata nel Backend Zoom;
Anche qui tutto il traffico di dati viene crittografato (anche grazie alla crittografia end-to-end se abilitata).
Riassumento dunque, il TOE riesce ad implementare tutte queste "features" che ne consentono un utilizzo sicuro:

- Autenticazione Sicura (anche per Meeting)
- Protezione della riservatezza e integrità dei dati trasferiti durante i meeting
- Applicazione di regole di controllo accessi durante i meeting
- Applicazione dei controlli utente
- Protezione della riservatezza, integrità e confidenzialità dei dati scambiati nelle Zoom Chat, inclusa protezione con crittografia end-to-end
- Protezione dei dati Zoom Chat memorizzati su dispositivo utente

Requisiti Non-Toe Hardware E Software

Per quanto riguarda le componenti hardware e software, il TOE non richiede particolari implementazioni di utilizzo. Il TOE deve essere eseguito su pc Windows o MacOS, o dispositivi mobile. Per i dispositivi android è solamento richiesto che il TOE venga eseguito su ambienti gestiti; questi dispositivi, poi, devono utilizzare un modulo di sicurezza integrato.
Infine, il ST consiglia l'utilizzo di dispositivi Android certificati come "consigliati" per le aziende, tra questi troviamo lo Xiaomi Mi 10t pro e il Google Pixel 4xl; questi dispositivi hanno un più elevato livello di garanzia.

Conformance Claims

Il Security Target reclama la conformità basata su Common Criteria (versione 3.1), nello specifico:

- Common Criteria Part 1: Introduction and General Model [CC1]
- Common Criteria Part 2: Security Functional Components [CC2]
- Common Criteria Part 3: Security Assurance Components [CC3]

come segue:

- CC Part 2 extended
- CC Part 3 conformant (EAL2)

Definizioni problemi di sicurezza

Il Security Target fornisce tutte quelle definizioni che permettono di analizzare e comprendere il quadro generale di tutti quelle definizioni (come minacce, assunzioni, assets) che serviranno poi ad organizzare gli obbiettivi di sicurezza del TOE.
Nelle definizioni di problemi di sicurezza fanno anche parte le Security Policy; nel ST viene definita una Policy di tipo organizzativa:

OSP.Meeting_Password_Policy : gli organizzatori (host) non devono scegliere una password più debole di quella generata automaticamente dal TOE per l'accesso ad un meeting

Assets

Gli assets sono tutte quelle informazioni, parametri o features che devono essere protette all'interno del TOE. La loro sicurezza è fondamentale per il corretto utilizzo del TOE.
Di seguito abbiamo alcuni assets importanti:

- **SA.User_Credentials** : dati di accesso associati ad un utente : username e password
- **SA.User_Data** : dati associati ad un utente come mail, nome, cognome, contatti, altre informazioni utente
- **SA.Meeting_Credentials** : dati di accesso associati ad un meeting : id riunione e password
- **SA.Meeting_Data** : dati trasferiti tra un utente e il backend Zoom durante una riunione : ad esempio flussi vocali, video, chat, file, dati condivisione schermo
- **SA.Meeting_Key** : chiave usata per cifrare la comunicazione durante un meeting
- **SA.User_Controls** : controlli scelti/usati dagli utenti durante un meeting, come attivazione audio/video, microfono ecc
- **SA.Zoom_Chat_Data** : dati trasferiti tra utente e Zoom Backend durante l'uso di Zoom Chat, come i messaggi di testo, voce e video
- **SA.Stored_Chat_Data** : dati Zoom Chat salvati nel dispositivo host dell'utente

Minacce

Le minacce sono tutti quei pericoli o possibili attacchi che devono essere evitati per far si che gli assets siano protetti, e di conseguenza anche la protezione del TOE. Di seguito si ha l'elenco di tutte le minacce che sono state identificate pericolose per il TOE:

- **T.Access_Login** : bypassare le funzioni di sicurezza del TOE per ottenere l'accesso non autorizzato a SA.User_Data e SA.User_Credentials
- **T.Access_Meeting** : accesso alla riunione in modo non autorizzato così da ottenere gli SA.Meeting_Data, SA.Meeting_Credentials e SA.Meeting_Key
- **T.Network** : dirottamento della comunicazione tra backend e TOE manipolando quindi un accesso a SA.Meeting_Data, SA.Meeting_Key, SA_Meeting_Data_to_Backend, SA.Zoom_Chat_Data
- **T.Bypass_User_Controls** : accesso, all'interno di una riunione, all'audio condiviso, fotocamera, dati condivisione schermo anche se SA.User_Controls è impostato per non trasmettere tali dati
- **T.Access_Stored_Data** : un attaccante ottiene l'accesso al dispositivo host del TOE e accede a SA.Stored_Zoom_Chat_Data

Assunzioni

Le assunzioni sono tutte quelle definizioni che ci si aspetta che il TOE faccia per far si che quest'ultimo sia sicuro

- **A.Meeting_Key** : il backend genera SA.Meeting_Key fornendo sicurezza di almeno 120 bit

- **A.User_Credentials** : ogni utente rivela le proprie credenziali o quelle di riunione a soggetti non autorizzati all'accesso del TOE o al meeting

- **A.Rate_Limiting** : il backend implementa la limitazione di veolcità su attacchi brute force per password sai per login utente che per login meeting

- **A.Secure_Backend** : il backend è affidabile; si presume che i dati inviati da esso siano integri

- **A.Host_Device** : il dispositivo host del TOE offre mezzi per archiviare e accedere in modo sicuro alle chiavi crittografiche, nonchè fornisce un generatore di numeri casuali di forza adeguata da utilizzare per operazioni crittografiche

- **A.Managed_Device** : il TOE viene eseguito su un dispositivo gestito; la gestione del dispositivo controlla quali applicazioni possono essere installate per garantire la sicurezza del TOE (evita l'installazione di app dannose)

- **A.Proper_User** : l'utente che utilizza il TOE non è intenzionalmente negligente od ostile e rispetta le politiche di sicurezza

- **A.Non_Hostile_Platform** : la piattaforma su cui gira il TOE non invia alcun attacco agli asset protetti dal TOE

Obbiettivi di Sicurezza

Gli obbiettivi di sicurezza sono quelle definizioni che identificano come contrastare o mitigare le minacce identificate dagli sviluppatori; inoltre devono seguire anche le politiche di sicurezza per il TOE, come quella organizzativa identificata precedentemente, e rispettare le assunzioni fatte.

- **O.Secure_Channel** : il collegamento tra TOE e backend deve essere un canale sicuro e affidabile in modo da proteggere la riservatezza e l'integrità delle informazioni trasmesse

- **O.Meeting_Encryption** : il TOE deve fornire una crittografia avanzata per tutti i dati trasmessi durante una riunione usando SA.Meeting_Key

- **O.Secure_Storage** : il TOE deve garantire che i dati Zoom Chat memorizzati sul dispositivo host del TOE non vengono mai archiviati in testo normale

- **O.Meeting_Authentication** : il TOE deve garantire che solo gli utenti autenticati possano accedere alla riunione

- **O.Authentication** : il TOE deve garantire che solo gli utenti autenticati possano accedere ai propri dati

- **O.User_Control** : il TOE deve applicare il SA.User_Controls per ogni partecipante del meeting

- **O.User_Control_Remote** : il TOE deve implementare il controllo degli accessi in modo tale che gli utenti possono usare controlli solo in base agli attributi di sicurezza impostati

- **O.Advanced_Chat_Encryption** : il TOE deve implementare chat crittografate end-to-end tra i membri della stessa organizzazione

Nel Security Target vengono anche definiti gli obbiettivi di sicurezza per l'ambiente operativo. Nel caso del TOE in analisi riguardano il dispositivo host dove viene eseguita l'applicazione , gli utenti e il Zoom Backend.

- **OE.Meeting_Password_Policy** : gli utenti che usano il TOE non devono indebolire la password generata automaticamente durante la creazione di un meeting

- **OE.Keys** : il backend deve generare SA.Meeting_Key fornendo sicurezza di almeno 120 bit

- **OE.User_Credentials** : gli utenti che usano il TOE non devono divulgare a possibili utenti non autorizzati

- **OE.Rate_Limiting** : il backend deve implementare la limitazione di

velocità dopo un certo numero di tentativo di autenticazione fallite per un utente o riunione

- **OE.Secure_Key_Storage** : il dispositivo host del TOE deve fornire i mezzi per archiviare e accedere in modo sicuro alle chiavi crittografiche

- **OE.Secure_Server_Storage** : il backend deve proteggere la riservatezza e integrità di tutti i dati inviati

- **OE.Meeting_Security_Attributes** : il backend deve memorizzare, proteggere e aggiornare gli attributi di sicurezza della riunione per gli organizzatori di riunioni autenticati; questi includono autorizzazioni ai partecipanti e l'identificazione dell'utente host riunione

- **OE.Meeting_Security_Controls** : il backend deve applicare i controlli di sicurezza impostati dal meeting tra host e co-host

- **OE.Password_Strengh** : il backend deve garantire che le password selezionate dall'utente siano complesse

- **OE.Managed_Device** : l'organizzaione che usa il TOE deve assicurarsi che quest'ultimo sia installato su dispositivi gestiti; la gestione del client deve garantire che non siano installate app dannose

- **OE.Proper_User** : l'utente dell'applicazione non devono essere negligente o malevole nell'utilizzare il TOE e quindi deve rispettare le policy di sicurezza

Gli obbiettivi di sicurezza sono definiti per far si che le minacce siano "coperte" e le assunzioni siano rispettate; in questo modo, si ha un quadro completo di tutte le definizioni di sicurezza del TOE. Naturalmente una volta definiti gli obbiettivi di sicurezza, questi devono essere raggiunti tramite i **Security Requirements e le Security Assurance**, ossia funzionalità e garanzie che permettono che gli obbiettivi siano soddisfatti.

Security Requirements

Nel ST vengono definiti soltanto i Security Functional Requirements: questi servono per specificare le singole funzioni di sicurezza che gli sviluppatori devono implementare per far si che coprono gli obbiettivi di sicurezza, e quindi fornire sicurezza al TOE.
Il TOE utilizza anche una componente estesa , la FCS_CKM_EXT.1 . L'uso della componente estesa è concesso in quanto il Security Target è conforme al

CC parte 2 con estensione di questa componente.

Functional Requirements

Di seguito vi sono alcuni degli SFR presi in analisi:

FIA UID.1

- 1.1 : il TSF deve consentire l'accesso e modifica delle impostazioni generali (incluse impostazioni predefinite per attivazione telecamera e audio) durante le riunioni per conto dell'utente prima che esso venga identificato
- 1.2 : il TSF richiede che ogni utente sia identificato con successo prima di consentire qualsiasi altra azioni mediata da TSF per conto dell'utente

FIA_UAU.1/client

- 1.2 : il TSF richiede che ogni utente sia autenticato con successo prima di consentire qualsiasi altra azione mediata da TSF per conto dell'utente

Queste due componenti, che riguardano l'autenticazione e identificazione, permettono di coprire l'obbiettivo **O.Authentication** : l'autenticazione richiede che il TOE autentica gli utenti prima che essi possano accedere ai loro dati utente.

FIA_UAU.2/Meeting

- 2.1 : il TSF richiede che ogni utente sia autenticato con successo prima di consentire qualsiasi altra azione mediata da TSF per conto dell'utente. Questo SFR è per l'autenticazione ad un meeting; la partecipazione è concessa anche tramite link d'invito

La componente copre l'obbiettivo di autenticazione di **O.Meeting_Authentication** : simile alle due componenti sopra, questa richiede che il TOE consenta solo all'utente autenticato di partecipare ad una riunione.

Per conto dei controlli degli accessi , abbiamo una delle seguenti componenti:

FDP_ACF.1/UC - controllo degli accessi su attributi

- 1.1/UC : il TSF applica la politica di controllo utenti sui partecipanti e su SA.User_Controls
- 1.2/UC : il TSF applica le seguenti regole per determinare se un operazione tra soggetti controllati e oggetti controllati è consentita : *“ il diritto di abilitare/disabilitare i controlli utenti è concesso in base gli attributi di sicurezza riunione”*
- 1.3/UC : il TSF autorizza esplicitamente l’accesso dei soggetti agli oggetti sulla base delle seguenti regole aggiuntive: *“ L’ospite di una riunione ha pieno accesso e può abilitare/disabilitare i propri controlli utente senza limitazioni”*
- 1.4/UC: il TSF nega esplicitamente l’accesso dei soggetti agli oggetti sulla base delle seguenti regole aggiuntive: *[nessuna]*

La componente consente di soddisfare l’obbiettivo **O.Secure_Control_Remote** : si richiede, infatti, che il TOE applichi le politiche di controllo degli accessi ai controlli utente in base agli attributi di sicurezza all’interno della riunione; queste regole sono anche spiegate dalle altre componenti FDP_ACC.1/UC e FMT_MSA.3/UC.

FMT_SMR.1 - gestione e ruoli di sicurezza

- 1.1 : il TSF manterrà i ruoli *utente autenticato, utente non autenticato, host, co-host e partecipante*
- 1.2 : il TSF deve essere in grado di associare gli utenti ai ruoli

FMT_SMF.1 - specificazione funzioni di gestione

- 1.1 : il TSF deve essere in grado di svolgere le seguenti funzioni di gestione:
    - Mutare/smutare microfono
    - Abilitare/disabilitare videocamera
    - Abilitare/disabilitare condivisione-schermo
    - Abilitare/disabilitare registrazione meeting

Anche la componente FMT_SMR.1 copre l’obbiettivo **O.Secure_Remote_Control** spiegando le politiche di controllo degli accessi.

La componente FMT_SMF.1 ,invece, descrive i controlli degli utenti

richiedendo al TOE di implementarli. Dunque va a soddisfare l'obbiettivo **O.User_Control**.

FTP_ITC.1 - trusted channel

- 1.1 : il TSF deve fornire un canale di comunicazione tra se e un altro prodotto IT affidabile che sia logicamente distinto dagli altri canali di comunicazione e fornisca garanzia
- 1.2 : il TSF deve consentire di avviare una comunicazione tramite canale attendibile
- 1.3 : il TSF deve avviare la comunicazione tramite canale attendibile per qualsiasi comunicazione tra backend ed app (eccezione del trasferimento dei dati riunione). Nota: il canale attendibile è una connessione TLS tra Zoom App e Zoom Backend

Questa componente va a ricoprire più obbiettivi di sicurezza per il TOE. Questi obbiettivi sono **O.Secure Channel** richiedendo al TOE l'implementazione di un canale di comunicazione sicuro; **O.Meeting_Encryption** perchè, oltre ad importare la chiave di un meeting su quel canale, quest'ultimo fornisce un ulteriore livello di protezione per la trasmissione dei dati crittografati; infine O.Advanced_Chat_Encryption sempre per lo stesso motivo descritto per l'obbiettivo O.Meeting_Encryption.

FCS.CKM.1/AES

- 1.1 : il TSF deve generare chiavi crittografiche in conformità con uno specifico algoritmo di generazione chiave simmetriche e di dimensioni specifiche di 256 bit che soddisfano quanto segue NIST FIPS 197[NIST FIPS 197]. Nota: le chiavi AES sono usate più volte sia per crittografare i dati Zoom Chat che il database locale

La seguente componente tratta i supporti crittografici. Dalla definizione si può dedurre che la componente va definire come viene generata la chiave per le operazioni crittografiche (algortitmo AES): la chiave poi dovra essere importata ed utilizzata all'interno delle riunioni, delle chat, e per archiviare i dati. Dunque soddisfa gli obbiettivi **O.Meeting_Encryption** , **O.Secure_Storage** e **O.Advanced_Chat_Encryption**.

La componente ha poi estensione che definisce il funzionamento dell'importazione delle chiavi.

FCS_CKM_EXT.1.1 : il TSF deve importare le chiavi crittografiche da un altra entità IT affidabile che può essere utilizzata per la crittografia AES-256.

L'importazione delle chiave deve essere fatta per una riunione per poi consentire la cifratura del traffico dati (quindi copre l'obbiettivo **O.Meeting_Encryption**)

L'ultima componente funzionale in analisi riguarda sempre l'aspetto di sicurezza dei dati, in questo caso le operazioni di cifratura di essi:

FCS_COP.1/AES-GCM - operazioni crittografiche

- 1.1 : il TSF deve eseguire la crittografia simmetrica in conformità con uno specifico algoritmo crittografico AES-GCM con dimensioni della chiave d 256 bit che soddisfano quanto segue: NIST FIPS 197[NIST FIPS 197] , NIST SP 800-38D [NIST SP 800-38D]

La componente delinea come il TOE deve eseguire le operazioni di cifratura durante un meeting così poi da consentire la trasmissione non in chiaro dei dati tramite un canale sicuro (copre l'obbiettivo **O.Meeting_Encryption**)

Gli SFR analizzati in esempio, ma in complesso tutti gli SFR definiti nel Security Target per il TOE, permettono di raggiungere gli obbiettivi di sicurezza definiti.

Assurance Requirements

I Security Assurance Requirements sono stati definiti secondo **EAL2** grazie alla conformità del Common Criteria descritte nella parte 3 [CC3].

EAL2 è stato scelto perchè fornisce una garanzia di sicurezza completa e un analisi degli Security Functional Requirements. L'analisi è supportata da test indipendenti del TSF, dai test eseguiti dagli sviluppatori basati sulle specifiche funzionali e da un analisi delle vulnerabilità, dimostrando una resistenza a penetration attackers e based attack.