YEO Messaging

Security Whitepaper

2022

Table of Contents

1 Introduction

YEO Secure Messenger system is a client-server technology that allows registered users to exchange messages using the YEO client app running on Android and iOS devices.

The client-server system architecture includes a client app (YEO client app) and a server (YEO server).

This document provides an overview of the implemented functionalities to deliver data confidentiality and privacy to the end-users.

2 User Registration

A user has to create an account by registering a few personal details. Registration is performed in two seamless steps using the YEO Client app and a valid mobile phone number.

2.1 User details registration

During the user details registration steps, the user has to provide a first name and last name, an email address, a valid and not-already registered mobile phone number and set a strong password according to the app password policy. The Mobile Phone number is verified through a six-digit code sent by SMS (short message service) from the YEO server to the user's mobile device.

The six-digit code is valid for one hour and the user can request a maximum of 3 verification codes in one minute. User details are securely stored on the user's device using strong encryption algorithms and secure key length.

The user's password is stored on the YEO server in a salted and hashed format. Password is not stored on the user's mobile device except when the user enables OS mobile biometric authentication protection to sign in to the YEO account.

User details are stored in the YEO server, which is encrypted at rest using AES symmetric encryption algorithm and a key length of 256 bit protected with Hardware Secure Module (HSM).

2.2 User face enrolment

For the second and final step in the registration process, the user enrols their face for YEO's continuous face recognition authentication. The face enrolment phase records the user's face to build a biometric tracker. Users can enrol one single biometric data (single user) during user registration. A user has the option to skip the face enrolment and verification process and complete it at a later stage. However, the user will need to complete the biometric enrolment to view or send YEO Mode messages (facial recognition required messages).

A biometric tracker is the binary representation of the user's face metric used in the face recognition process.

The biometric tracker is encrypted and securely stored locally on the user's device to provide continuous offline face recognition authentication.

During normal app usage, the YEO face recognition engine constantly updates the biometric tracker data, "learning" the user's day-by-day face changes. Biometric data is collected and stored on the user's device only.

2.3 Cryptographic material

YEO client app implements end-to-end message encryption using asymmetric cryptography to securely establish shared encryption sessions between users and to authenticate encrypted data and key material, and symmetric cryptography to encrypt messages.

Asymmetric cryptographic key pairs are created during the registration process.

Public keys data

is sent to the YEO server for storing in the YEO database server and it's available to other registered users to initialise the end-to-end encryption session with the user.

Private keys data

is securely stored in the user's device and it never leaves the user's device.

The YEO server is not involved in the message end-to-end encryption as it doesn't store or manage any user's private key.

2.4 Further data

During the registration, the YEO client app collects and stores in the YEO database server the following data:

- User profile photo of the user's face, matching the enrolled face during the face recognition enrolment process unless the user decides to skip the biometric enrolment.

This data allows the user to provide more information to other YEO users at specific identity-evaluation times, for example when a user is presented with a YEO user profile contact request.

The user profile photo is encrypted using AES symmetric encryption algorithm and 256-bit key length. The encryption data is stored on the YEO server using another layer of encryption at rest using AES and 256-bit key length protected with Hardware Secure Module (HSM).

Users can delete the entire account and metadata through the YEO client app functionality at any moment.

3 Authentication

3.1 API end-points

All communications between YEO client apps and the YEO server are encrypted using Transport Layer Security (TLS) 1.2 and Perfect Forward Secrecy (PFS) based cypher suites.

After a successful login, the YEO client app is granted authorisation to access the YEO server API end-points using a refresh token system where the access token has a very short lifetime and allows the server to revoke the tokens easier in case abuse is detected.

Authorisation tokens are digitally signed using the Elliptic Curve Digital Signature algorithm and securely stored on the user's device on iOS and Android.

YEO client granted tokens are invalidated when the user signs out using the YEO client app signing-out functionality.

3.2 Sign-in

A user can sign in to the YEO account by providing:

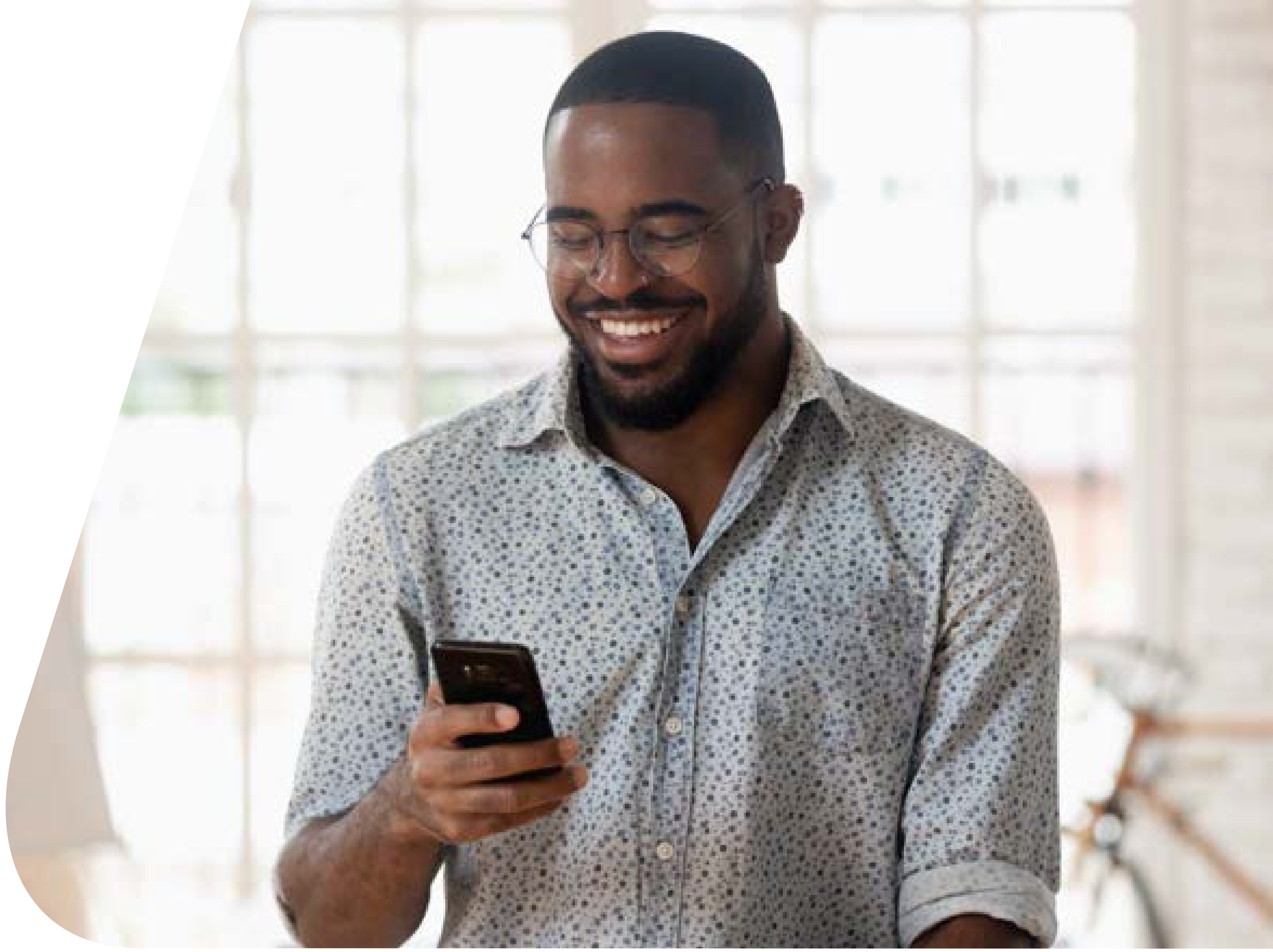
- Registered and verified mobile phone number
- Password

After a successful sign-in the API end-points authorisation is renewed and saved on the user's device.

Users can enable OS biometric authentication in the YEO client app settings **(5.1 Mobile Biometric Authentication)** to sign in to the YEO account without the need to type the YEO account password.

3.3 Sign-out

Users can manually sign out at any time through the YEO client app ***sign-out*** functionality. Once the sign-out process is completed the server invalidates the granted API end-points authorisation tokens.



3.4 Password reset

YEO client app provides a password reset functionality. To reset the password the user provides the registered mobile phone number to receive a six-digit verification code through SMS service, the six-digit verification code is valid for one hour.

On a successful code verification, the user can set a new password according to the password security policy. Once the password is reset the YEO client app is authorised to access the API endpoints using new authorisation tokens.

There can only be one pending password reset for an account and a new password reset cannot be initiated until the previous password reset process timeout.

4 Messaging

4.1 Contact

A user can send a contact request to another user for which the requester has the mobile phone number already in the device contacts list. The recipient can accept the request, ignore it or report it as spam.

To increase user privacy, users can avoid receiving indirect and unwanted contact requests by changing the account visibility in the app settings.

When the user sets the account visibility to a hidden state, no one of the other users can view the public profile using the client app contact functionalities, even if they have the user's mobile phone number in the device contact list. This protects the user from unwanted and indirect contact requests.

4.2 Message security features

To protect sender privacy, received messages can’t be forwarded.

The YEO app also blocks screenshotting on Android devices. On iOS devices, due to Apple restrictions, the YEO app notifies the sender when a recipient has taken a screenshot and the app immediately deletes all received messages on the recipient side, preventing the iOS user from taking further screenshots.

The sender can delete sent messages and attachments from the recipient device as soon as the recipient YEO client app is online.

Users can apply the following conditions/restrictions to the messages to enhance the protection of the confidentiality of the message content:

Geofence

Sender can set a geographical circle area where the message can be viewed by the recipient.

Burn-after-reading

Sender can set a viewing timeout to auto delete the messages when the timeout expires. A deleted message is removed from the recipient device.

Face Recognition authentication

Sender can enforce recipient continuous face authentication to view a message.

4.3 One-to-one messaging

YEO secure messenger encrypts exchanged messages and attachments in a one-to-one conversation with end-to-end encryption.

End-to-end encryption guarantees that only the sender and the intended recipient are able to read the messages, preventing potential eavesdroppers.

YEO end-to-end encryption uses an implementation of the Triple Diffie-Hellman (3DH) key exchange algorithm [1] to securely establish a shared encryption session between two users without transmitting secret cryptography key material. Triple Diffie-Hellman implementation uses elliptic curve cryptography with curve25519 elliptic curve and X25519 Diffie-Hellman algorithm.

Once the shared encryption session is established on both ends an implementation of the Double Ratchet Key Management algorithm [2] is used to manage message encryption keys directly on the sender and recipient devices without transmitting the message encryption keys.

Messages are encrypted using the per-message derived encryption key from the double ratchet algorithm and using the symmetric algorithm AES-256 in CBC operation mode and the encrypted message authenticated with HMAC-SHA256.

Please refer to the "YEO Cryptographic Whitepaper" [3] for more details on the Triple Diffie-Hellman and Double Ratchet implementations.

Message attachments are encrypted using the symmetric algorithm AES-256 in CBC mode HMAC-256 authenticated. The 256-bit encryption key is randomly generated from a secure random generator and it’s encrypted and delivered to the recipient in an end-to-end encrypted message.

The key management algorithm provides the following security properties:

Per-message key

Each message is encrypted with a different key, mitigating key-leak attacks. A compromised key decrypts only a single message and not new or old messages.

Lost messages

Messages not delivered doesn't affect future messages.

Future secrecy (Post-Compromise Security)

To automatically re-establish secrecy in case the user's long-live identity private cryptographic material is compromised

Out-of-order decryption

Encrypted messages can be decrypted when delivered in a different order from when they have been encrypted.

Forward secrecy

Each encrypted session uses an ephemeral key exchange mitigating session key-leak attack. A compromised session key decrypts only past messages for a specific encrypted session until the Future Secrecy feature re-establish the security of the session.

4.4 Group messaging

YEO supports end-to-end encrypted group conversation using a client fan-out model, a sent message is end-to-end encrypted specifically for each recipient's device, enforcing a secure and unique one-to-one encryption session per user's device.

Using the same implementation of the Triple Diffie-Hellman key exchange algorithm, the Double Ratchet key management algorithm and AES-256 symmetric cryptographic algorithm for message encryption used in the one-to-one message exchange, a group conversation has the same level of security and the same security properties of a one-to-one conversation.

The user creating a group conversation is automatically the group administrator and can add and remove users or transfer administrator status to another member of the group.
The administrator is the only user in a group with the privileges to modify the group state and details.

4.5 Message metadata

Message metadata are data created and related to a sent or received message. Metadata for messages are securely stored on the YEO database server encrypted at rest using AES symmetric encryption algorithm and 256-bit key length.

Message metadata includes:

- Message id
- Message receiver id
- Message timestamps
- Message recipient device id
- Message sender id
- Conversation id

4.6 Message deletion

Messages are always deleted from the YEO server after that they are successfully delivered to the recipient device. Users can delete an already sent message at any time. An already delivered message is deleted from the recipient's device as soon as the recipient is online. Server and client delete the message body (text and attachments) and the message metadata.

A deleted message cannot be recovered.

4.7 Message storage

The YEO client app enforces security and user privacy by storing sent and received messages and attachments on the user’s device encrypted with AES symmetric algorithm and an encryption key length of 256-bits and HMAC-SHA256.

4.8 Message delivery

Like any YEO client app communication, YEO end-to-end encrypted messages are sent and delivered using the Tunnel Layer Security (TLS) protocol, which establishes an encrypted communication tunnel between the YEO server and YEO client app. For security and compliance reasons, YEO supports only TLS 1.2 version and Perfect Forward Secrecy strong cypher suites.

4.9 Notifications

When the client app is not running, it's not connected to the YEO Server end-points but it still receives message and contact related notifications using the Apple and Google external push services.

To enforce privacy and security, YEO push notifications don't transport confidential information in the payload so they don't reveal any information when they appear on the user's mobile home screen.

To provide a smoother user experience, the YEO Client app is able to receive messages and attachments when it is in the background and the user doesn't directly interact through the app user interface.

5 YEO Biometric Security

YEO client app implements biometric security to authenticate the user at several points during the app execution.

Two types of biometric security features are used:

- Mobile OS biometric authentication
- YEO continue face recognition authentication

5.1 Mobile OS biometric authentication

Mobile OS biometric authentication functionalities are available only on devices that provide biometric capabilities, i.e. fingerprint scanner or face recognition.

User can enable mobile OS biometric authentication functionalities to enhance the app security for:

- YEO account biometric sign-in
- YEO biometric screen unlocking

YEO account biometric sign-in allows the user to sign in to the YEO account without typing the account password. When this functionality is enabled, the mobile OS provides further security through strong hardware encryption and secure storage.

YEO biometric screen unlocking functionality provides additional protection when the client app moves from foreground app-state to background app-state and vice versa. When a user opens the client app again from the background app state, the YEO client app requires the user to authenticate using the mobile OS biometric functionality.

5.2 YEO continue face recognition authentication

YEO client app provides continuous face recognition to continually authenticate the user during message viewing. To protect the privacy a sender of a message can enforce face recognition on the recipient side for specific and confidential messages.

Continuous face recognition prevents an unrecognised and unauthenticated user from viewing the message content of a privacy protected message. The content of the privacy protected message is immediately obfuscated when the user is not face authenticated.

Enforcing continuous face authentication for message viewing is a per-message feature that can be activated on a specific single message when needed or enabled and automatically applied to multiple messages.

6 YEO App Code Hardening

YEO client app implements code hardening techniques to prevent the use of the app on compromised mobile operating systems and to make code analysis and reverse engineering more difficult. A compromised or tampered mobile operating system not only allows a malicious user to analyse and modify the running apps but also introduces system-wide vulnerabilities by disabling system built-in security controls.

YEO client implements countermeasures to prevent, detect and mitigate the most common app binary attacks including dynamic and static analysis.

YEO client app also implements an encryption layer for all data at rest along with the operating system built-in file system encryption at rest. A double data encryption layer at rest protects the YEO app's data when the device operating system security controls, like app sandboxing, is compromised.

7 Compliance

YEO secure messenger system is compliant with:

- EU General Data Protection Regulation (GDPR 2016/679)
- UK GDPR / UK Data Protection Act 2018
- UK ICO