Tungsten Labs UG (haftungsbeschränkt)
Email: contact@tungsten-labs.com
Web: http://tungsten-labs.com
Monbijouplatz 5,
10178 Berlin

Tungsten Security Whitepaper

Berlin, May 2018

Version 1

Contents

Introduction

This document gives a technical and security overview of the Tungsten messenger applications and their protocol. All messages and data sent via Tungsten are end-to-end encrypted. Tungsten uses its own Double Ratchet protocol, with support for offline messaging, multi-device setups as well as group messaging.

Tungsten allows users to create multiple identities to keep their conversations isolated. With multiple identities, users can easily manage who can contact them, without compromising their personal information or mixing up conversations.

Additionally, Tungsten offers an easy way to set up another device with access to the complete message history (see the chapter on Tungsten Sync for more details). This requires physical access to both devices, which adds an additional level of security.

Table 1 Feature comparison for Tungsten and other popular protocols.

Feature	Open PGP	OTR	Tungsten
Multiple devices	Yes	No	Yes
Offline Messages	Yes	Yes	Yes
File Transfer	Yes	No	Yes
Verifiability	No	Yes	Yes
Deniability	Yes	Yes	Yes
Forward Secrecy	No	Yes	Yes
Future Secrecy	No	Yes	Yes

Personas

Types of Personas

Tungsten allows each user to have two types of personas: public and anonymous. Our core philosophy is that a user should be able to assume multiple personas that aren't linked to any personal information such as a phone number, email address or even the IP addresses that a user connects from. Anonymous personas always connect through Tor. This means that the extra encryption and obfuscation that Tor provides, is present until your connection reaches Tungsten's infrastructure. The end result is an anonymous persona which is incredibly difficult to trace back to any real world identity.

Table 2 Feature comparison for public and anonymous personas.

Feature	Public persona	Anonymous persona
All messages end-to-end encrypted	Yes	Yes
Send messages via Tor	No	Yes
Can be added by username	Yes	Yes
Can be added by Address Book synchronization	Yes	No
Can be transferred via Tungsten Sync	Yes	Yes
Use 3rd party service, like push notifications	Yes	No

Tungsten doesn't store any personal information for anonymous personas, except the data needed to allow the system to operate. Additionally, no third-party solutions such as notifications service are used to protect the users' anonymity (see the chapter on Notifications for more details).

Multiple Personas

Users are able to have multiple personas logged in simultaneously on the same device. Those accounts don't share any data since this is stored in separate, local databases. Users are limited to having one public and multiple anonymous personas. It's not required to create a public persona in order to set up a anonymous one. Currently, multiple anonymous personas logged in on the same device share the same Tor circuit. We plan to change this in the future.

Persona Registration

Each user can select which type of persona should be registered first: public or anonymous. Additional personas can always be added later from within the app.

When registering a public persona, the user must provide a display name and an unique username. Additionally, a valid phone number needs to be set, which is then used by Tungsten to send a verification code via SMS gateway. This phone number is also used to login on any other devices, and serves as an identification method so that other Tungsten users can find the user.

To register an anonymous persona no phone number or email is needed. The only data required during this form of registration, is a display name, unique username and a password.
When creating an anonymous persona, users should keep in mind that due to the general visibility of usernames and display names, any private information can lead to a compromise of anonymity. For this reason, users should consider using Tungsten's built-in name generator when signing up.

Metadata

Tungsten uses a minimum set of unencrypted metadata to operate.

- Username
- Display Name
- Phone number (when provided)
- Persona Avatar
- Conversation Name
- Conversation Avatar

For future releases we are aiming on making Persona Avatar, Conversation Name and Conversation Avatar encrypted as well.

Multi-Device Security

Tungsten users can use the same persona on multiple devices at the same time, regardless of the supported platform. Each new device requires prior registration in the system to obtain unique identifier.

Tungsten Sync

Having multiple devices or changing phones are common scenarios nowadays. Since all data stored on the server is kept only until it is delivered and is encrypted for every device, downloading a message history is impossible. Tungsten Sync is an alternative for users to migrate local data from one device to another without uploading it to the Cloud. Due to different platform capabilities, our current implementation only supports migration between iOS and macOS. Android support is planned for future releases.
The synchronisation process is split in two phases: *Initiation* and *Transfer*. The transfer process is device type independent, initiation differs between mobile and desktop device types.

Initiation

In order to perform synchronisation, a secure session has to be established between both devices. Tungsten sync uses Multipeer Connectivity API for local communication, so Bluetooth and WiFi could be used. Internet connection is only required for authorization purposes, user data is not sent over the Internet during this process.
QR codes are used to exchange data needed to setup transfer session. This code contains a session identifier advertised by Source Device (**SD**) as well as its public key and protocol version. A Destination Device (**DD**) scans the code and connects to the session advertised by **SD**.

Transfer

1. Both devices establish a connection and enter synchronisation mode
    - **DD** sends *public_key* to **SD**
    - Both devices calculate *encryption_key* using Diffie-Hellman key exchange & SHA256
    - **SD** sends list of Personas encrypted using *encryption_key* and AES256 method
2. User on **DD** selects personas to synchronise
3. **SD** contacts Tungsten backend to obtain temporary *transfer_authentication_tokens* for each selected persona
4. **SD** prepares *transfer_bundle* and encrypts it with an *encryption_key* and AES256 method
5. **DD** processes transfered data and obtains *long_lived_access_token* from Tungsten backend

Contacts

To communicate, each Tungsten persona has to connect with other personas. Those connections are Contacts.

Contact Requests

In order to add someone as a contact, the username has to be known and used to send a Contact Request. The person receiving the request can choose to accept, reject or ignore it. Until the request is accepted either side cannot communicate directly. Both sides can communicate indirectly when participating in same group conversation. A contact request can be cancelled at any time.

Contact Removal

Each persona can remove any existing contact. When a contact is removed from the contact list, the *1-on-1* conversation history is also deleted. If the removed contact is part of a group conversation, those conversations are kept and remain active.

Contact Synchronization

Tungsten allows users to find other users present in the system by synchronizing their own Address Book. Each phone number entry in an address book is formatted following ITU's E.123 and E.164 recommendations, and is then hashed and sent to the server in order to perform matching. Each matched contact from the address book is automatically marked as a contact of that persona. This synchronization is performed only for public personas and only if synchronisation was enabled. Address book hashing happens whenever Tungsten is opened or the user switches to this persona. When matching is completed on backend, all Address Book hashes are removed from the server. New contacts will only be added when the next synchronization is performed by the client.

IMPORTANT: Contact synchronization is possible for Public Personas only. Doing so would compromise the anonymity of a Anonymous Persona, since knowing someone's contacts list allows to assume their identity to a certain degree. Users should be extra vigilant when inviting contacts while using Anonymous Persona.

Messaging

End-to-End Encryption

All messages and attachments exchanged via Tungsten are end-to-end encrypted, meaning no one, not even someone with the access to the Tungsten Server, can see the content of your messages. Tungsten's end-to-end encryption protocol is inspired by Double Ratchet algorithm. The main features of this protocol are:

- Each message and its attachment is encrypted using a different AES Message Key, which is derived from the key of the previous message.
- Multi device support - AES Message Key in encrypted separately for every target device in the chat (both sender and receiver device)
- Forward Secrecy - breaking one message key doesn't allow to decrypt the previous messages
- Message Reordering - receiving messages in different order than they were sent doesn't impact communication
- Offline support - receiver's device doesn't have to be online to setup the cryptographic session

Table 3 Security algorithms.

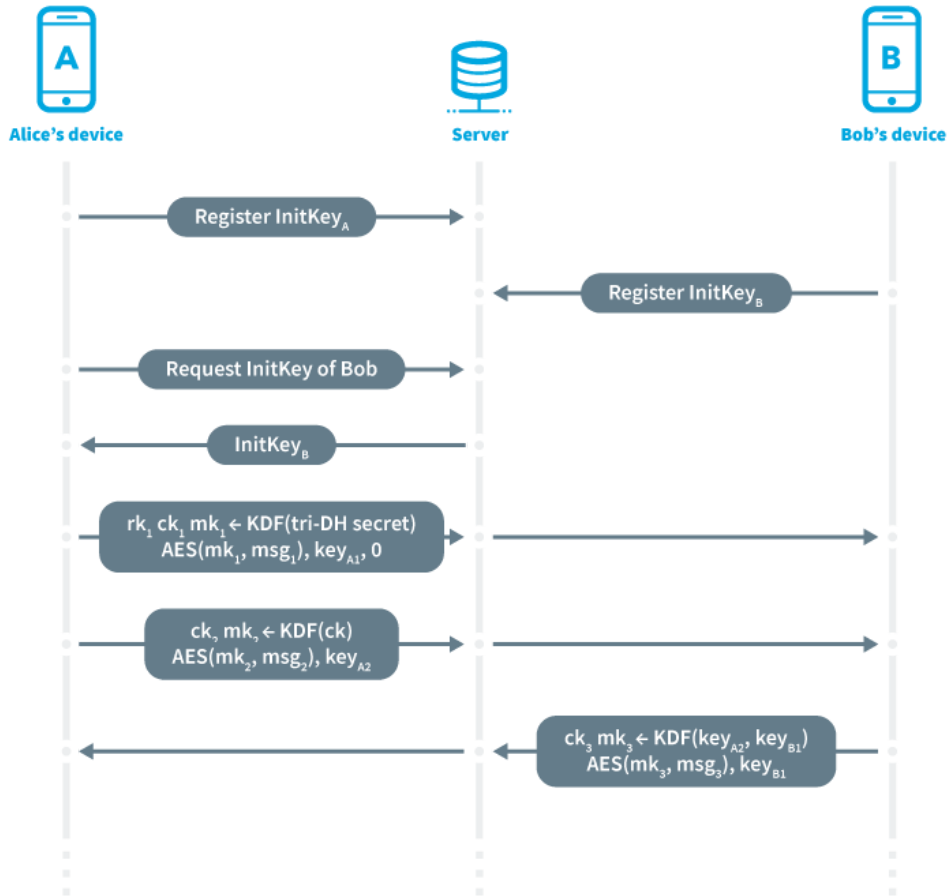
Security component	Algorithm	Android implementation	iOS and macOS implementation
Symmetric encryption	AES 256 with GCM	Bouncy Castle	OpenSSL
Key Derivation function	HKDF	Libsodium	Libsodium
Message Authentication Code	HMAC SHA 256	Bouncy Castle	CommonCrypto
Diffie-Hellman key exchange	Curve25519 and Ed25519	Libsodium	Libsodium
AES key generation	Secure random	Libsodium	RAND_bytes, OpenSSL

Session Initialisation

In order to be able to establish a cryptographic session between any two devices, a Diffie-Hellman key exchange algorithm is used. As one of the devices might be offline during the DH key exchange, the following algorithm was used:

- During registration, each device generates a set of Init Keys and stores them in the local storage. Each Init Key consists of a public and private key.
- Each device uploads the public portion of the Init Key Bundle to the Tungsten Server.
- In case if Alice wants to establish session with Bob, Alice downloads the Init Key uploaded by Bob and establishes the cryptographic session using this key. When Bob receives the first encrypted message from Alice, he uses the private part of the Init Key to establish and verify the session.
- Init Key infrastructure is used only to establish the session, every subsequent message is derived from the actual session state and doesn't require the Init Keys.

Figure 1 presents a detailed schema of the communication between Alice and Bob, described above.



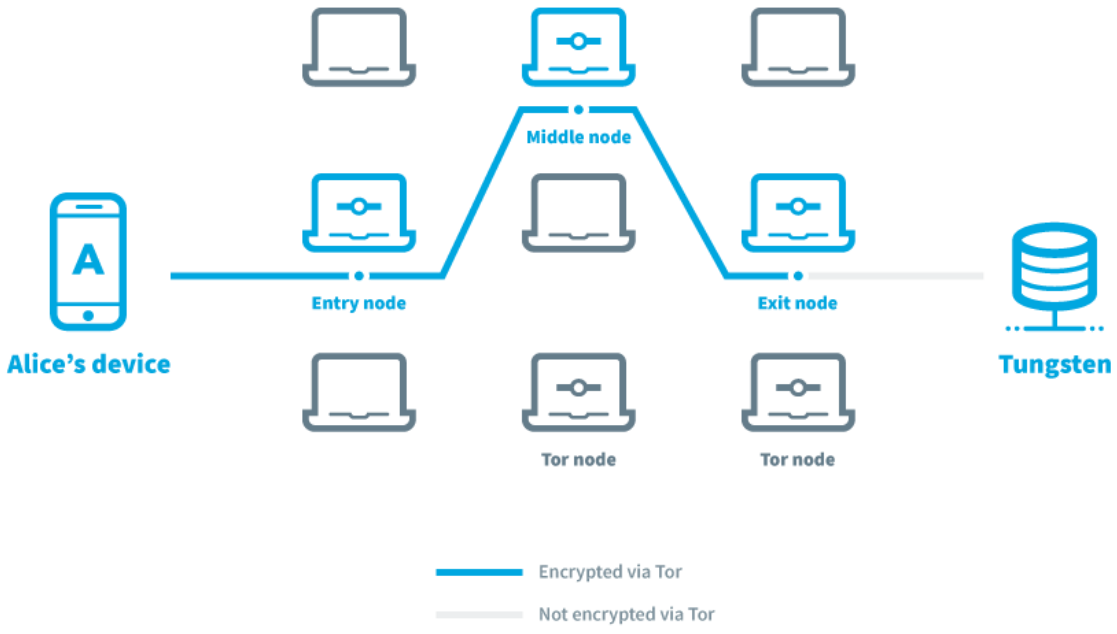
Message Retention / Storage on Backend

Messages are kept on server until delivered to recipient device. As soon as delivery is confirmed by client application, copy is deleted from database on server.

Tor

Anytime we use the internet, our devices leave traces leading back to us. Standard traffic analysis can be used to detect who is talking to who over a public network. We believe that privacy is not only about encrypting message content, but also anonymising who you are in touch with.
This is why using a anonymous persona in Tungsten allows true anonymity by routing all network communication via Tor. This means that traffic is not monitored, and neither Tungsten or anyone else can access or store users' metadata.
Every single network request made by an anonymous persona will go through Tor. Which means every time users send messages, receive files or open links inside the Tungsten application their anonymity is protected.

Figure 2 End-to-end connection via Tor



Message Types

Tungsten allows users to send various types of messages, such as text, photo, video, audio, and documents. The system also contains a technical type of messages, which are used to inform the user about events in conversations, such as a new contact added, or that conversation was renamed etc.

Large File Upload

Large files are sent between users as media messages and are also end-to-end encrypted. When sending media messages the sender:

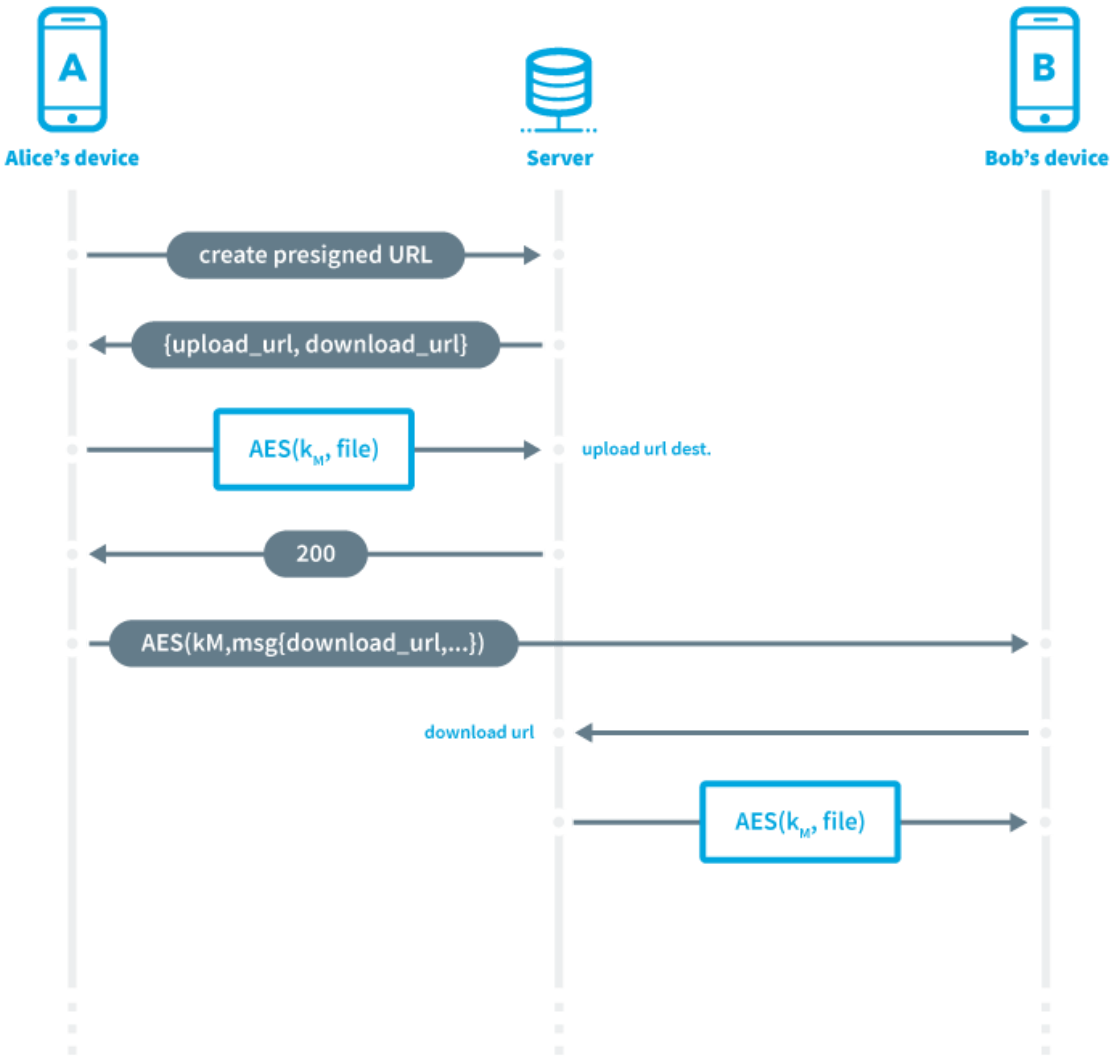
1. Generates a random *symmetric_key* for use with AES-256
2. Encrypts the file data with *symmetric_key* using GCM mode
3. Sends the encrypted file to Tungsten storage
    a. For images or video files an additionally encrypted thumbnail is sent
4. *symmetric_key* together with media metadata and download URL is encrypted for each recipient using Tungsten Crypto session
5. Media message is sent and distributed to all recipients

Each recipient of media message will:

1. Decrypt message using Tungsten Crypto session, thus obtain *symmetric_key* and all metadata needed to download file ciphertext
2. Download file ciphertext and decrypts it using *symmetric_key*

Only recipients present in the same conversation are able to receive messages from each other. Files are persistently stored on Tungsten storage without any timeout. This allows for multiple downloads of the same file and preserves local storage space on client device.
Forward secrecy for media messages are not affected since *symmetric_key* is secured using Tungsten Crypto session.

Figure 3 Large files sharing between Alice and Bob.



Notifications

Tungsten notifies users when the following events occur:

- a new message was received
- a user persona logged in on a new device (unless Tungsten Sync was used)
- a user's contact logged in on a new device
- a contact request was received
- a contact request was accepted or rejected

Notifications are only used to inform the user that certain events have occurred. Notifications don't contain any sensitive or private data such as message content.

Public Persona Notifications

When users use a public persona, system notification services are used. For Android, this is Firebase Cloud Messaging and for and iOS Apple Push Notification Service.
Only the data that is needed by those services to deliver notifications is shared: notification types, plain text containing public metadata and persona or conversation identifiers. No ciphertext is sent, since this could be stored and used for malicious purposes.

Anonymous Persona Notifications

For anonymous personas there's a strong focus on anonymity, so we opt not to use 3rd party notifications. Registering a user's device to Apple Push Notification Service or Firebase Cloud Messaging would compromise a persona's identity by storing these details on their servers. Instead, Tungsten's own '*notification queues*' have been implemented. Each persona contains a list of pending notifications that are fetched periodically in background while using the Tor network. This can result in some delays compared to public personas. Anonymous persona notifications also don't include plain text message content, which is displayed in the notifications bar. Instead, the server returns only specific meta-fields (such as notification type and e.g. persona or conversation identifier), relying on client applications to construct notification body.

Conversations

Tungsten identifies 2 types of conversations, *1-on-1* and *group*.

1-on-1

There can be just one of this type of conversation between two personas. Participants can't change conversation title nor avatar. Deleting any conversation will propagate to all involved devices.

Group Conversation

Group conversations can accommodate up to 20 participants at a time. Users can have as many group chats as they wish.
Each participant of the conversation is allowed to:

- rename the group
- change the group's avatar
- leave the group
- delete their personal conversation history
- invite people to the group
- remove any user from the group

In the future, we plan to introduce participant roles such as Administrator and User. The group's creator will be set as an administrator by default. Planned user permissions are defined below.

User permissions:

- rename the group
- change the group avatar
- leave the group
- delete their personal conversation history

Administrator permissions:

- rename the group
- change the group avatar
- invite people to the group
- remove any user from the group
- delete their personal conversation history
- delete the conversation for everybody

Client Security Overview

Data Storage

All data stored by the application, such as user profiles, contacts and communication history, are stored in an encrypted database.

For iOS, core Data Protection with Complete Protection policy is in use. For Android, the local storage is encrypted using an AES256 key. This key is protected inside the native Android Key Store. Due to an issue on older Android versions (Android L and lower), which wipes all the Keystore data in case the user changes a lock type, Tungsten uses Android Keystore only for devices running Android M and higher.

Logs

Logs don't contain any sensitive data. These logs are never automatically uploaded anywhere. Users have the option to attach them when contacting our support. This reduces any potential risk of data leakage via this channel.

Server logs contain basic information such as request params, response status as well as additional metadata, like response or broadcast time. Additionally, encrypted text (i.e. message payload, message keys data) is not stored at all. All log entries are removed 24 hours after being saved.

Platform Security

Root or jailbreak detection doesn't trigger any special events in the Tungsten application. There is a plan for a feature which informs users that the used device is rooted or jailbroken.