More is Less: On the End-to-End Security of Group Chats in Signal, WhatsApp, and Threema

Cosmin-Ionuț Rusu

About

Title:

More is Less: On the End-to-End Security of Group Chats in Signal, WhatsApp, and Threema

Authors:

Paul Rosler, Christian Mainka, Jorg Schwenk

Published Date:

January 15, 2018

Summary

- Proposed a **comprehensive** and **realistic** security model for group communication protocols
- Analyse three protocols: **Signal**, **WhatsApp**, and **Threema**
- Revealed:
  - Communication **integrity**
  - **Group closeness**
- Are **NOT** end-to-end protected
- In Signal, **Future Secrecy** does not hold for group communication

Vocabulary

- **Forward** and **Future Secrecy**: preservation or recovery of security if user secrets are leaked to the attacker at a **later** (resp. **earlier**) point of time
- In the context of people communicating in a room:
  - **Traceable delivery**: everyone in the room hears the communication
  - **Authenticity**: everyone knows who spoke
  - **No duplication**: how often words have been said
  - **No creation**: nobody outside the room can speak
  - **Confidentiality**: nobody outside the room can hear the communication inside
  - **Closeness**: the door is only open for invited persons

Assumptions and notations

- **Centralized** IM protocols:
  - **Central server** that receives, caches, and forwards messages from senders to receivers as soon as receivers are **online (asynchronously).**
- User stores **session state** containing **housekeeping variables** and **secrets** for the exclusive usage in the group.

$$gr = (ID_{gr}, \mathcal{G}_{gr}, \mathcal{G}^*_{gr}, info_{gr}), \mathcal{G}^*_{gr} \subseteq \mathcal{G}_{gr} \subseteq \mathcal{U}$$

$$\Sigma = ((\mathrm{snd}, \mathrm{rcv}), \\ (\mathrm{SndM}, \mathrm{Add}, \mathrm{Leave}, \mathrm{Rmv}, \mathrm{DelivM}, \mathrm{ModG}, \mathrm{Ack}))$$

Threat model

- **Malicious User** - a user that do not follow the protocol. Assume the members of the **target group** behave correctly.
- **Network Attacker** - full control over the communication network, may access and modify all unprotected traffic (not end-to-end protected)
- **Malicious Server** - attackers with access to the group instant messaging protocol
- **Long-term Secret Compromise** - obtain long-term secrets of a user
- **Session State Compromise** - obtain the full session state of a user

Security Goals

- **Confidentiality**
  - **End-to-end confidentiality**: no send messages can be obtained by the adversary
  - **Perfect Forward Secrecy**: on leakage of session state, confidentiality of past messages holds
  - **Future Secrecy**: renew session state and invalidate old ones
- **Integrity**
  - **Message Authentication**
  - **No creation**: only group members can send messages to the respective group
  - **No duplication**: a message is delivered at most once
  - **Traceable delivery**: if a member is notified about the termination of an action performed in the group, then the respective delivery was invoked by all its members.
  - **Weak FIFO order**
  - **Weak causal order**

Security Goals

- **Confidentiality by Group Management**
  - **Additive Closeness**: only group administrators can add new members
  - **Subtractive Closeness**: only group administrators can remove existings members, or a member can leave the group
- A protocol is a **Secure and Reliable Group IM protocol** if it fulfills E**nd-to-End Confidentiality**, **Message Authentication**, **No Creation**, **No Duplication**, **Traceable Delivery**, **Additive Closeness**, and **Subtractive Closeness** in the presence of Malicious User, Network Attacker, and Malicious Server.
  - **Bonus**: Perfect Forward Secrecy, and Future Secrecy
  - **Bonus (reliability)**: Ordering (trade-off instant message delivery)

Comparison between Protocols

Signal

- Open source
- Curve25519 and HMAC-SHA256 for key derivation (Double Ratchet algorithm)
- All members are admins
- Gr msg = Many direct msgs + gr id
- Gr updates encrypted
- ACKs rely on TLS only
- Members cannot be removed

WhatsApp

- Closed source
- Signal protocol for key exchange and encryption
- Group updates rely only on TLS (not e2e encrypted)
- Can have multiple **admins**
- Server responsible for group management
- Gr msg = one message to server with group id
- ACKs rely on TLS only

Threema

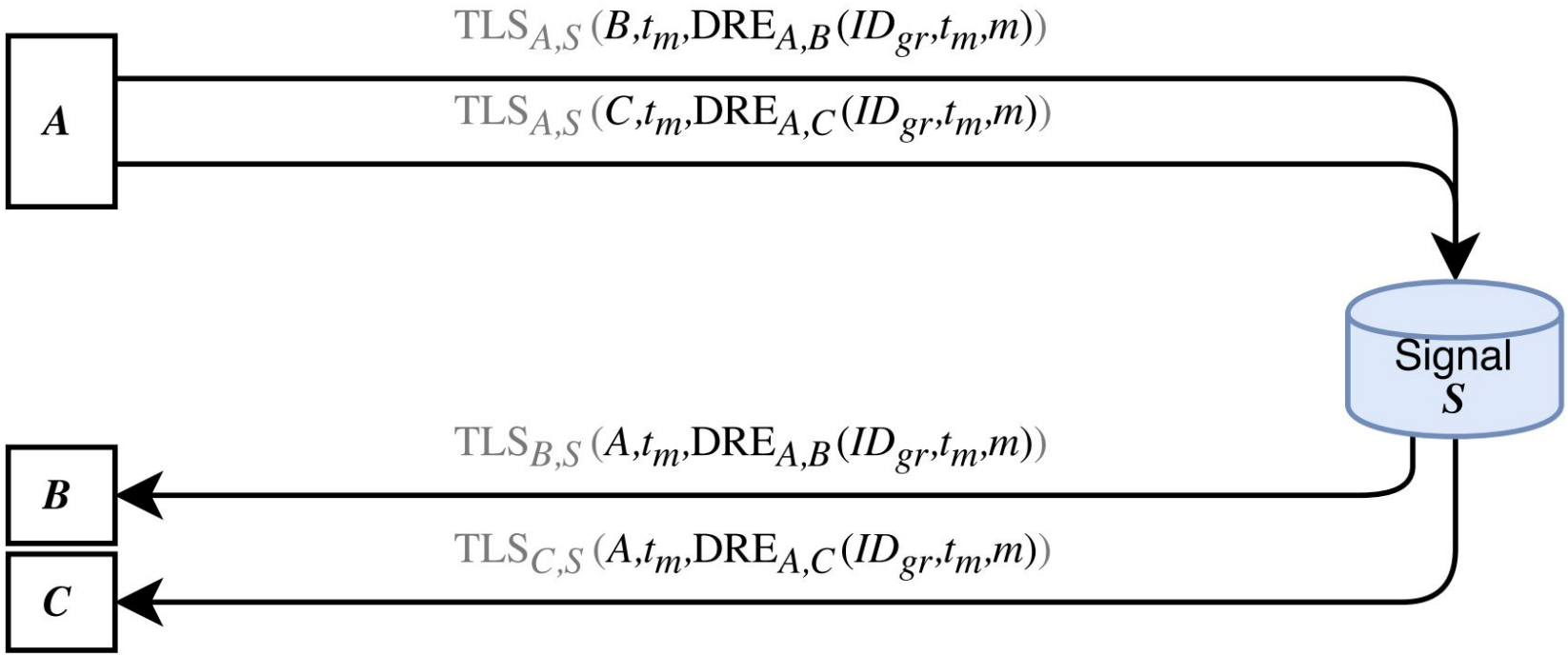
- Closed source
- **Long term** key (no key derivation)
- Creator is the only admin
- Gr msg = Many direct msgs + gr id
- No ACKs (does not support ‘seen’ feature)
- Admin can sync the group members

Signal

Signal protocol

1. **General Initialization Protocol**
   1.1. Session Establishment with the Server
   1.2. Key Agreement and Key Derivation
2. **Group Protocol**
   2.1. Group Messages
   - Treated as direct messages (server cannot distinguish between them)
   - Acknowledgements are **NOT** end to end encrypted and rely only on **TLS**

   2.2. Group Management
   - Update flow - sends the update group information (new set of members, and new group metadata) - **cannot** remove a person
   - Leave flow - sends the leave message with encrypted gid to all the other members


$\mathrm{TLS}_{A,S}\,(B,t_m,\mathrm{DRE}_{A,B}(ID_{gr},t_m,m))$
$\mathrm{TLS}_{A,S}\,(C,t_m,\mathrm{DRE}_{A,C}(ID_{gr},t_m,m))$
A
Signal
S
$\mathrm{TLS}_{B,S}\,(A,t_m,\mathrm{DRE}_{A,B}(ID_{gr},t_m,m))$
$\mathrm{TLS}_{C,S}\,(A,t_m,\mathrm{DRE}_{A,C}(ID_{gr},t_m,m))$
B
C

Signal protocol Security Evaluation

- **Burgle** into the target group by writing group management messages into it
  - **Preconditions:**
    - **Group id**
      - **Malicious User**: former member of the group that recorded the id
      - **Session State Compromise**: get the group id from the session state
    - **Phone number** of one member (B) of the target group
  - **Flow:**
    - The attacker A sends update group information message to B with him as a group member
    - B will gladly update it's group information locally and as soon as he makes another update operation on the group, he will add A. -could also send msg, but only B will get
    - Group will see that B added A (which is fine since B was a group member and hence an admin)
    - A is now an admin as well (all members are admins in Signal)

Signal protocol Security Evaluation

- Make a **victim** believe a message **is delivered**, while **it is not**
  - **Preconditions:**
    - **Malicious Server**: Deliver a message to a victim. Either compromise the server or bypass the transport layer protection
  - **Description**:
    - User B sends message to group
    - Attacker intercepts the message, drops it, and send ACKs back from each group members
    - B will think that all the group members received the message, while in fact no one saw that message.
  - **Impact**:
    - Violates **Traceable Delivery**
  - Not only messages can be dropped, but they can also be **reordered**

WhatsApp

WhatsApp Protocol

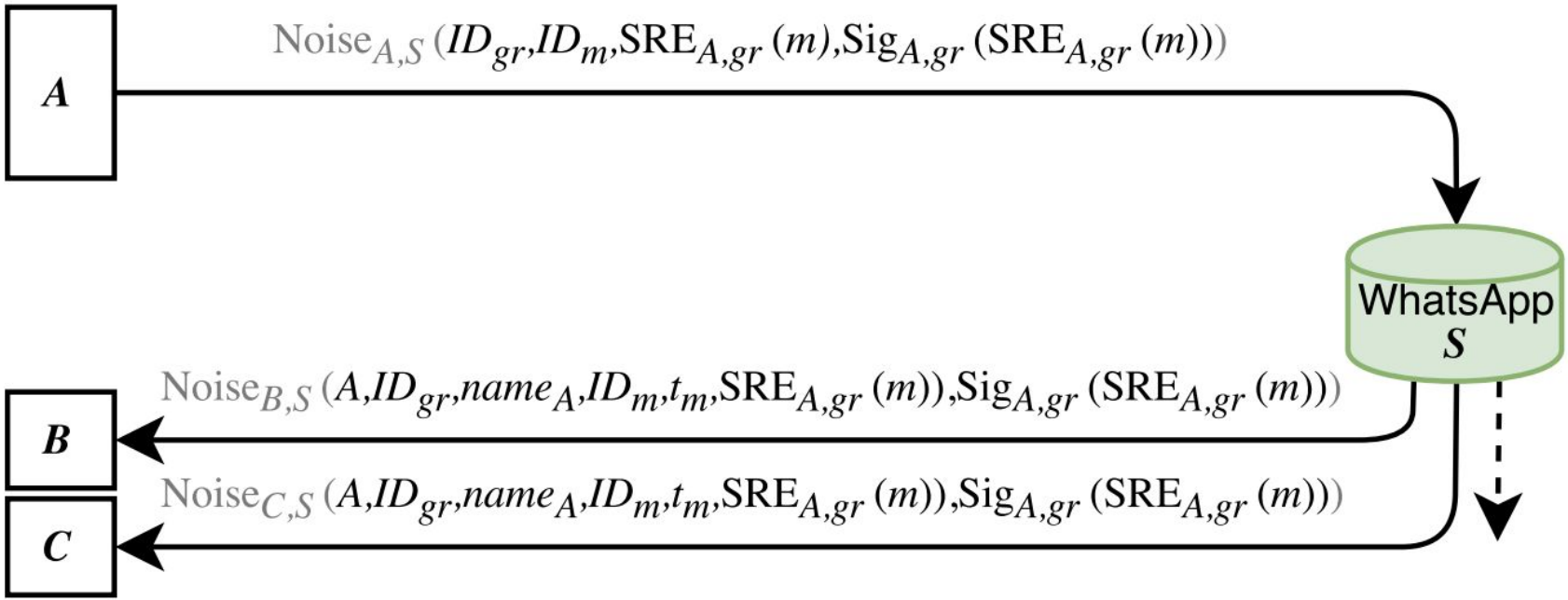
WhatsApp

- Closed source
- Uses **Signal** protocol for key exchange and encryption, **but different** group messaging/group managing protocol

WhatsApp Protocol

- Number of users limited to 256
- Content of messages are end-to-end encrypted, while group modification messages are only protected on the transport layer
- Server is responsible to for the distribution of group messages (compared to Sigma and Threema)

1. **Group content messages**
    a. **Sender** sends the group id, message id; **server** adds the sender id, sender name and timestamp
    b. **ACKs** are only protected on the transport layer
    c. Optional reference to a previous message (reply-to)
2. **Group management**
    a. **Only** protected on the transport layer
    b. Sends a tuple (action, H) where action indicates the operation type, and H - the affected users


A
$\text{Noise}_{A,S}\,(ID_{gr},ID_m,\text{SRE}_{A,gr}\,(m),\text{Sig}_{A,gr}\,(\text{SRE}_{A,gr}\,(m)))$
WhatsApp
S
B
$\text{Noise}_{B,S}\,(A,ID_{gr},name_A,ID_m,t_m,\text{SRE}_{A,gr}\,(m)),\text{Sig}_{A,gr}\,(\text{SRE}_{A,gr}\,(m)))$
C
$\text{Noise}_{C,S}\,(A,ID_{gr},name_A,ID_m,t_m,\text{SRE}_{A,gr}\,(m)),\text{Sig}_{A,gr}\,(\text{SRE}_{A,gr}\,(m)))$

WhatsApp Protocol Security Evaluation

1. **Burgle into a Group**
    a. **Preconditions:**
        i. Attacker **A** needs to **modify** the group information at the client side
        ii. **Malicious Server** can send group modification messages to the group members
    b. **Description:**
        i. Suppose a group with B, C, and D where B is the admin
        ii. Attacker sends group update ADD({A}) to C and D, with sender as B
        iii. Receivers send their keys
    c. Breaks **Additive Closeness**

WhatsApp Protocol Security Evaluation

2. Forging ACKs

a. **Preconditions**
   i. Attacker can drop messages and send notifications to senders
   ii. Only transport layer encrypted => **malicious server** can manipulate transcript between sender and receivers

b. **Description**
   i. Attacker drops a group message from the sender
   ii. Replies with ACKs from all the members

c. Breaks **Traceable Delivery**

Since the Group messages are only secured on the Transport Layer, allows an attacker controlling the WhatsApp server or the Transport Layer to take full control of the server. GUI displays **updates** => attacker can also **reorder** messages.

Threema

Threema protocol

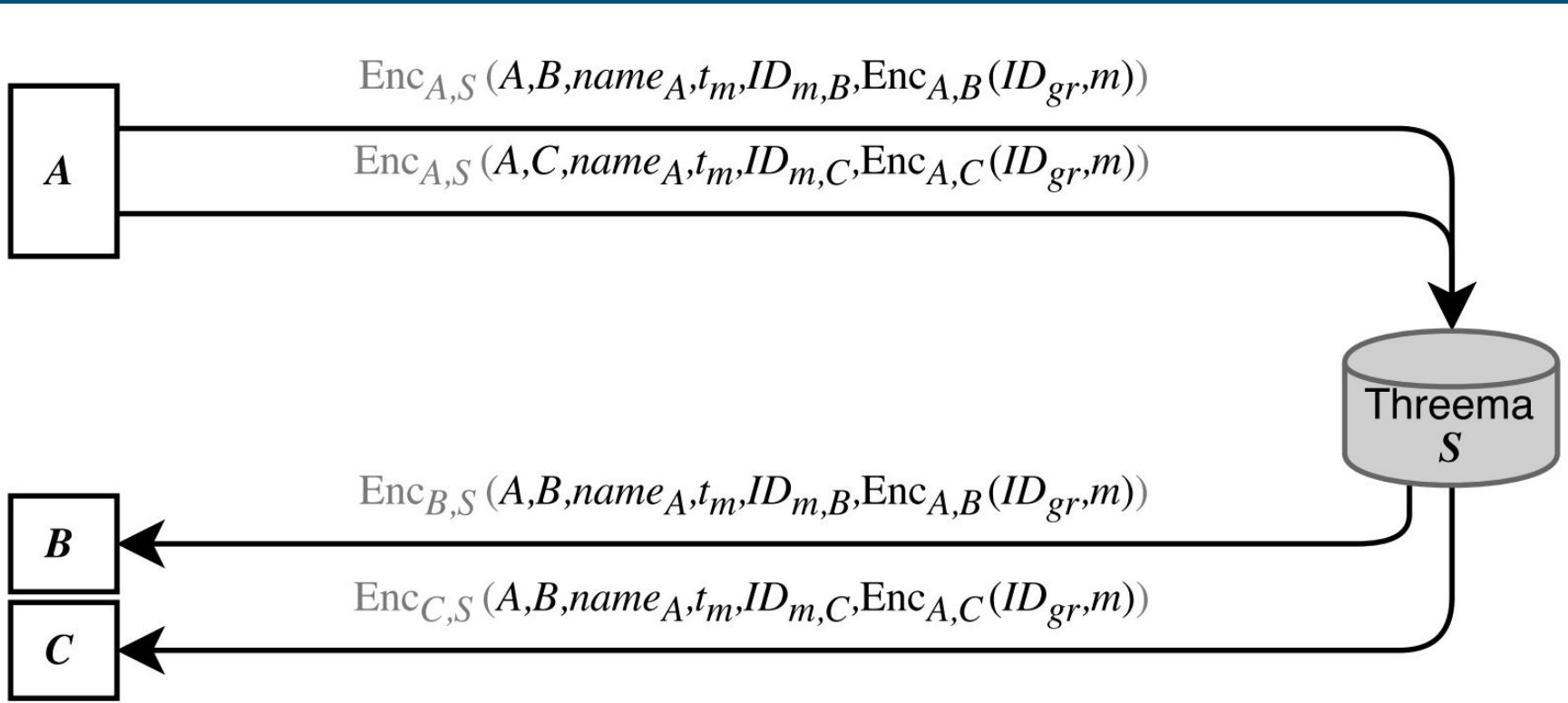
1. Group Protocol

1.1. Group Messages

- Treated as direct messages (server cannot distinguish between them)
- **No** receipt status (seen)
- Can have references to other messages (reply to)

1.2. Group Management

- **Creator** is the only admin
- Update flow - sends the update group information (new set of members, or new group metadata)
  - Group members can request the admin to sync group info
- Leave flow - sends the leave message with encrypted gid to all the other members


$\mathrm{Enc}_{A,S}\,(A,B,name_A,t_m,ID_{m,B},\mathrm{Enc}_{A,B}\,(ID_{gr},m))$
$\mathrm{Enc}_{A,S}\,(A,C,name_A,t_m,ID_{m,C},\mathrm{Enc}_{A,C}\,(ID_{gr},m))$
A
Threema
S
$\mathrm{Enc}_{B,S}\,(A,B,name_A,t_m,ID_{m,B},\mathrm{Enc}_{A,B}\,(ID_{gr},m))$
$\mathrm{Enc}_{C,S}\,(A,B,name_A,t_m,ID_{m,C},\mathrm{Enc}_{A,C}\,(ID_{gr},m))$
B
C

Threema Protocol Security Evaluation

1. **Replaying Messages**
    a. **Preconditions**
        i. Attacker needs access the channel between sender and receiver
        ii. **Malicious Server** has control over the transmitted ciphertexts
    b. All direct messages are encrypted and decrypted with the same key (no key derivation)
    c. **Description**:
        i. Record an encrypted message
        ii. Since only group id and the actual content is e2e encrypted, attacker can update the timestamp (and all other metadata) and replay the encrypted message
    d. Impact:
        i. **No duplication**: A can replay messages
        ii. **Additive Closeness**: A can rewind every group manipulation by resending previous group update messages.

Threema Protocol Security Evaluation

2. Protocol does not support **Traceable Delivery** (by the way it's designed)
3. No **Forward and Future Secrecy** due to encryption with the same key
4. **Ordering**: timestamp is not end-to-end encrypted => **reordering** is possible
5. **Additional Information Leakage: If user send a message he is not part of, the admin does a sync and responds to the user with the group members and info**

Thanks!