Man In The Contacts - Where Trust in Secure Messengers Leads to Spear Phishing

Swiss Cyber Storm

30/10/2018 – Securing Apps

whois securingapps

- Developer background
- French who spent last 12 years working in Switzerland on security products and solutions
  - Focus on mobile since 2010
- Now software security consultant at my own company

https://www.securingapps.com

- Provide services to build security in software
  - Mobile
  - Web
  - Cloud
  - Internet Of Things
  - Bitcoin/Blockchain

@SecuringApps

Introduction

- Popular messaging apps recently switched to End-to-End encryption
  - Great communication around it
  - Privacy now is a requirement
- Debates at the government level to ask for backdoors
  - Going dark ?
  - Used by terrorists ?
- Increased feeling that those applications are unbreakable

THE secure channel in companies

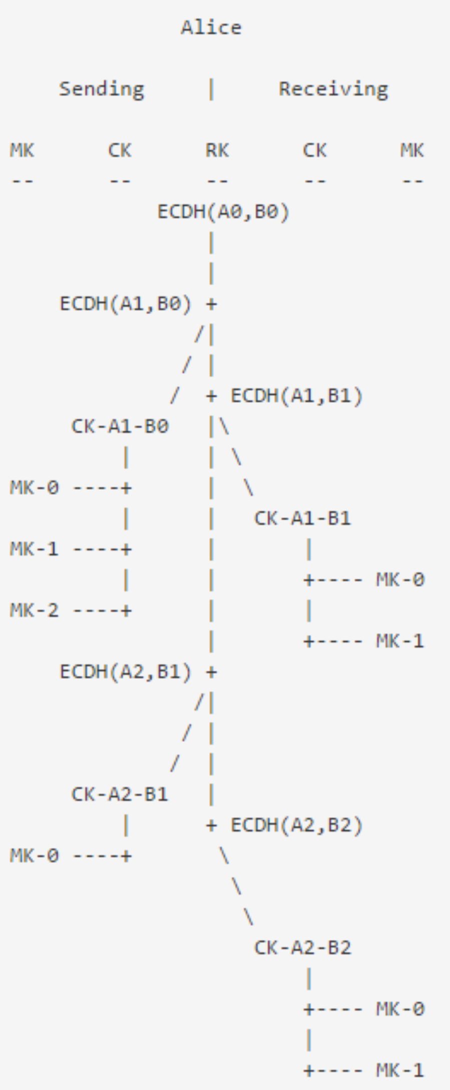
- Sharing temporary passwords
- Sending pictures with confidential data
- Discuss top secret topics rather than by email or by phone
- Fast priority channel
- And you don’t experience spam (yet)

Super crypto. But wait ….

- Advanced ratcheting in Signal Protocol →
- Looks like an obvious flaw won’t be there

- But how messaging apps authenticate myself ?
  - Provisioning done via SMS
  - Link to device/phone number

- And my contacts ?
  - Get them automatically from my address book



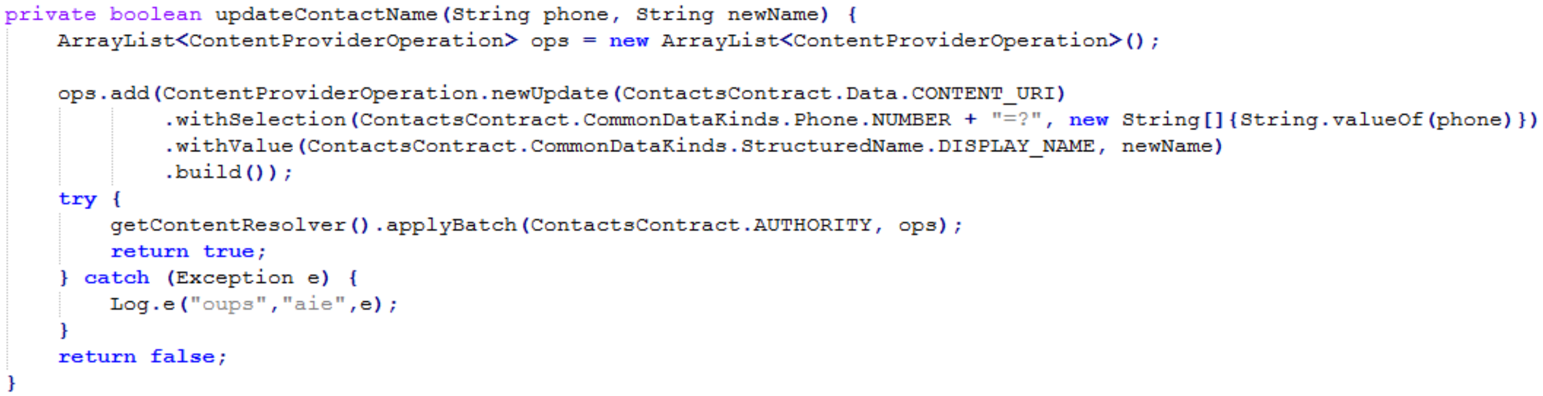
```
              Alice

    Sending     |     Receiving

MK      CK      RK      CK      MK
--      --      --      --      --
            ECDH(A0,B0)
                |
                |
    ECDH(A1,B0) +
               /|
              / |
             /  + ECDH(A1,B1)
     CK-A1-B0   |\
         |      | \
MK-0 ----+      |  \
         |      |   CK-A1-B1
MK-1 ----+      |       |
         |      |       +---- MK-0
MK-2 ----+      |       |
                |       +---- MK-1
    ECDH(A2,B1) +
               /|
              / |
             /  |
     CK-A2-B1   |
         |      + ECDH(A2,B2)
MK-0 ----+       \
                  \
                   \
                    CK-A2-B2
                        |
                        +---- MK-0
                        |
                        +---- MK-1
```

Accessing contacts

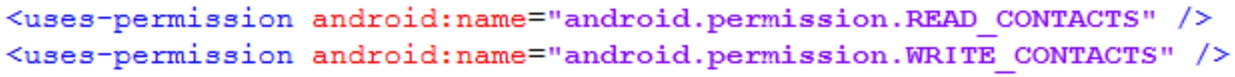
- Easy to read/modify/create contacts
  - There is an API for that
  - Android example



```java
private boolean updateContactName(String phone, String newName) {
    ArrayList<ContentProviderOperation> ops = new ArrayList<ContentProviderOperation>();

    ops.add(ContentProviderOperation.newUpdate(ContactsContract.Data.CONTENT_URI)
            .withSelection(ContactsContract.CommonDataKinds.Phone.NUMBER + "=?", new String[]{String.valueOf(phone)})
            .withValue(ContactsContract.CommonDataKinds.StructuredName.DISPLAY_NAME, newName)
            .build());
    try {
        getContentResolver().applyBatch(ContactsContract.AUTHORITY, ops);
        return true;
    } catch (Exception e) {
        Log.e("oups","aie",e);
    }
    return false;
}
```

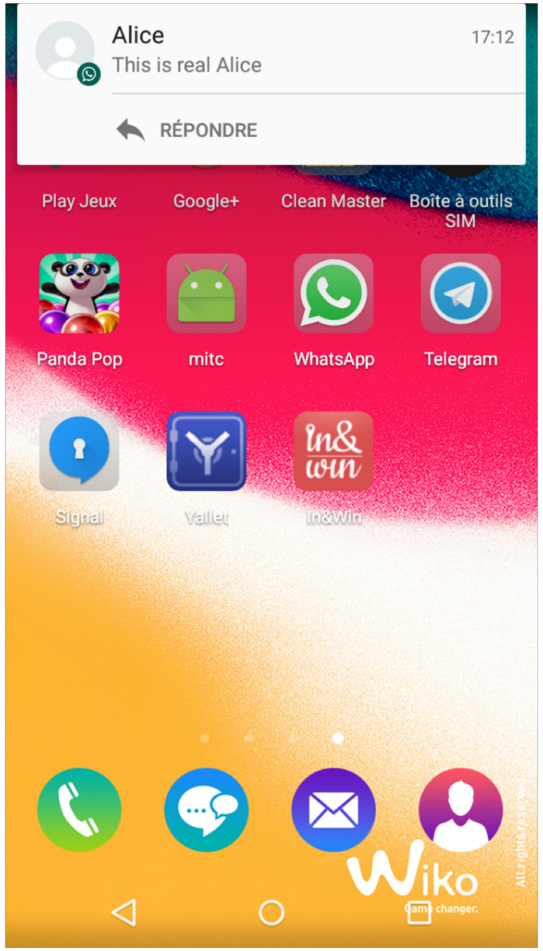
- Shared data structure accessible in read/write
  - Only restricted by permissions



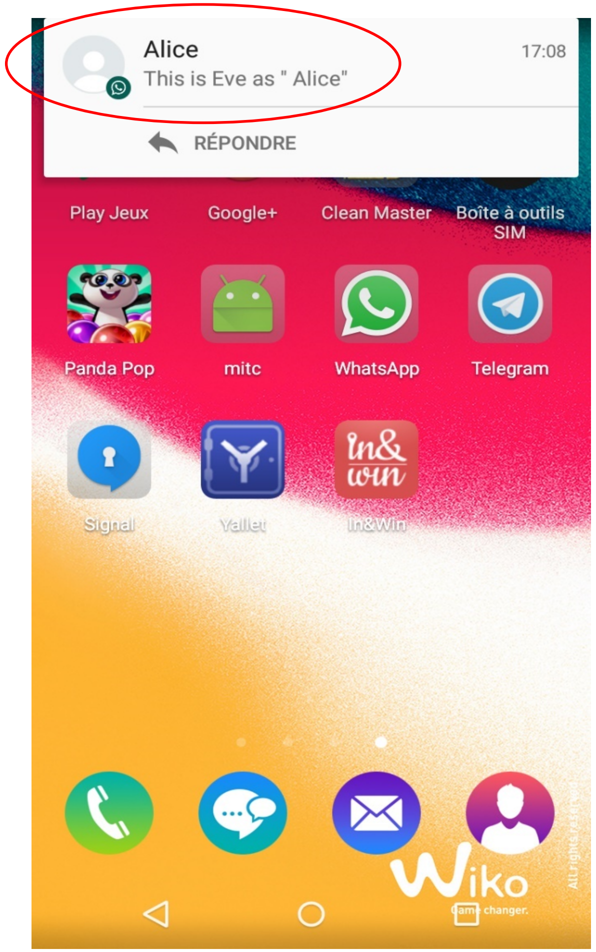
```xml
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
```

- There is **room for a side channel attack: Man In The Contacts**
  - **Not requiring a rooted device**

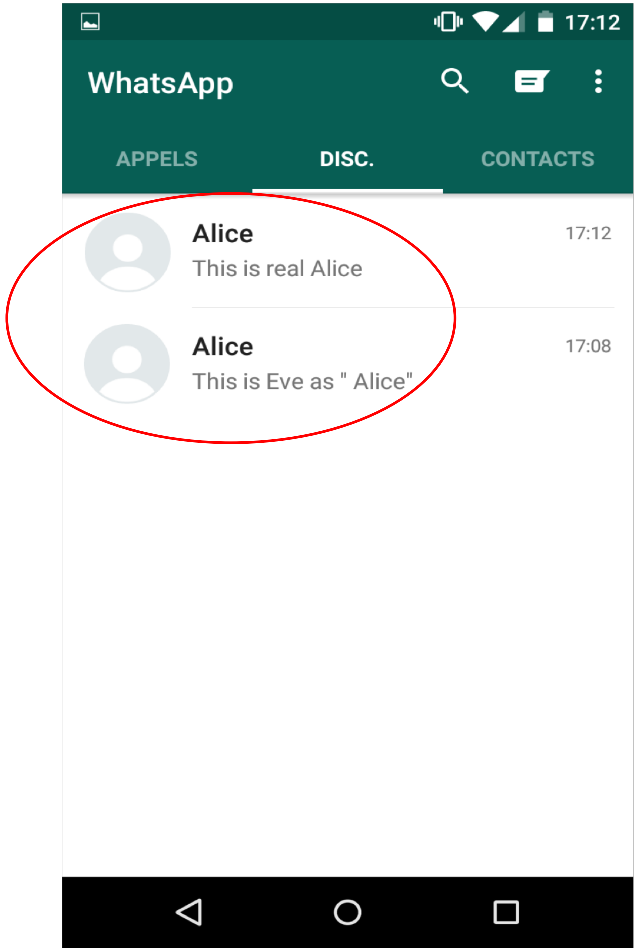
Let’s create a new contact « Alice»



Alice 17:12
This is real Alice



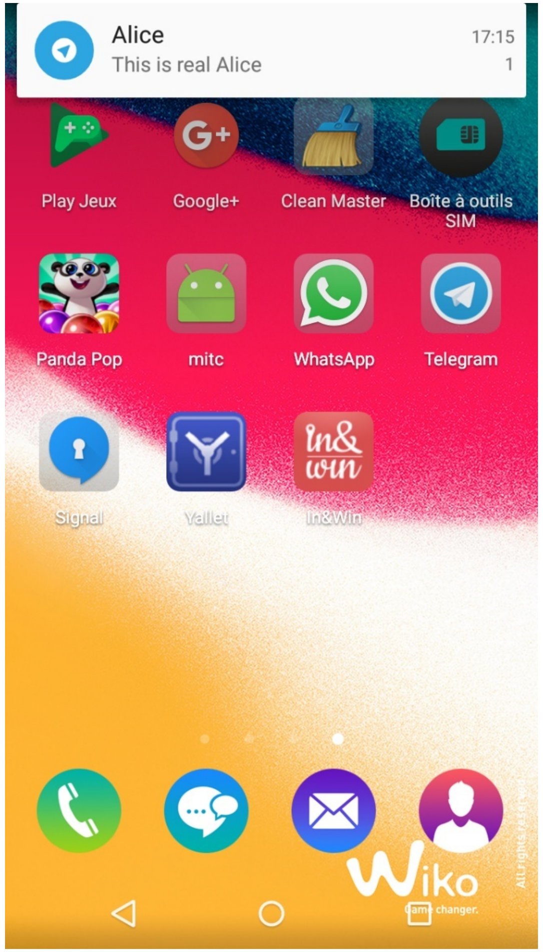
Alice 17:08
This is Eve as " Alice"



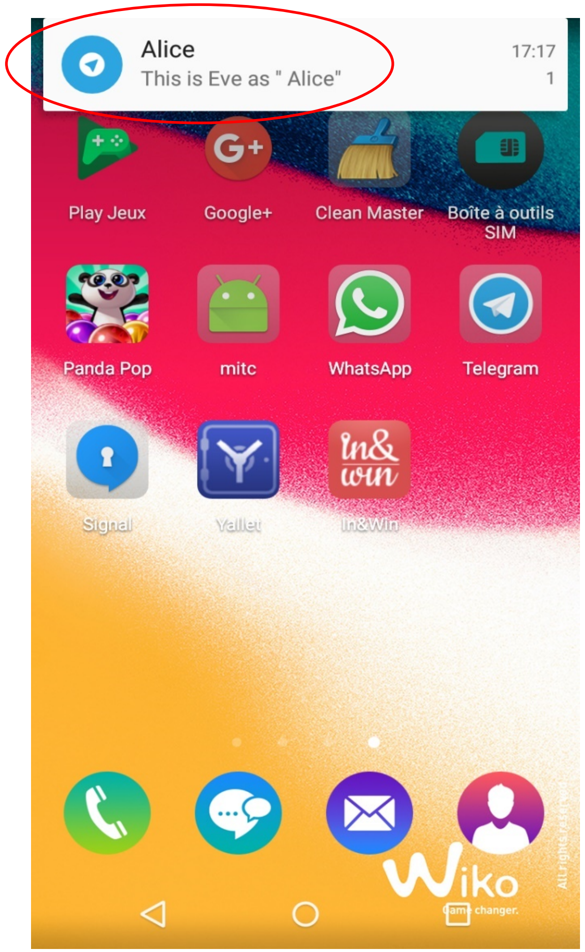
Alice 17:12
This is real Alice

Alice 17:08
This is Eve as " Alice"

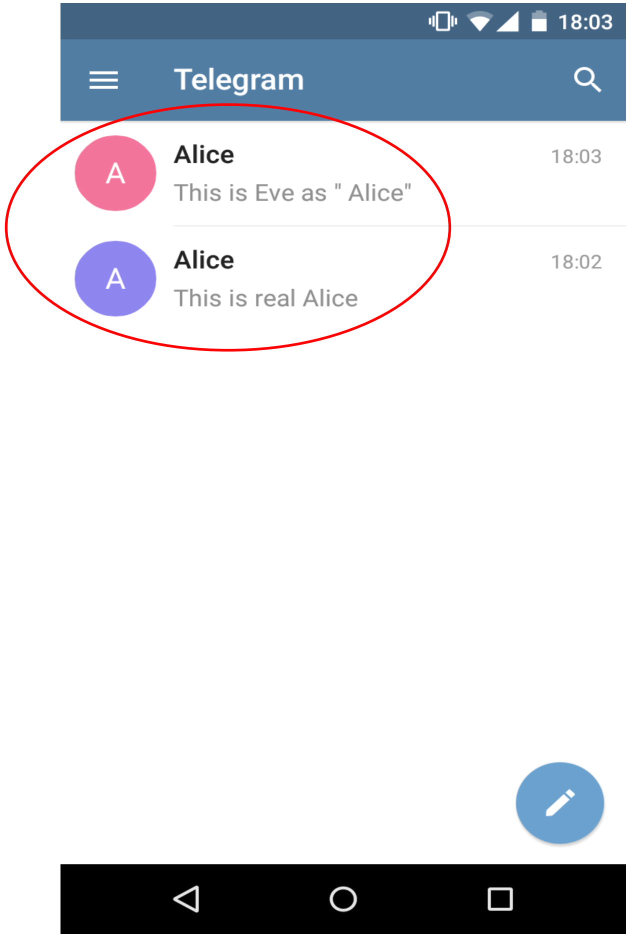
Let's create a new contact « Alice»



Alice 17:15
This is real Alice 1



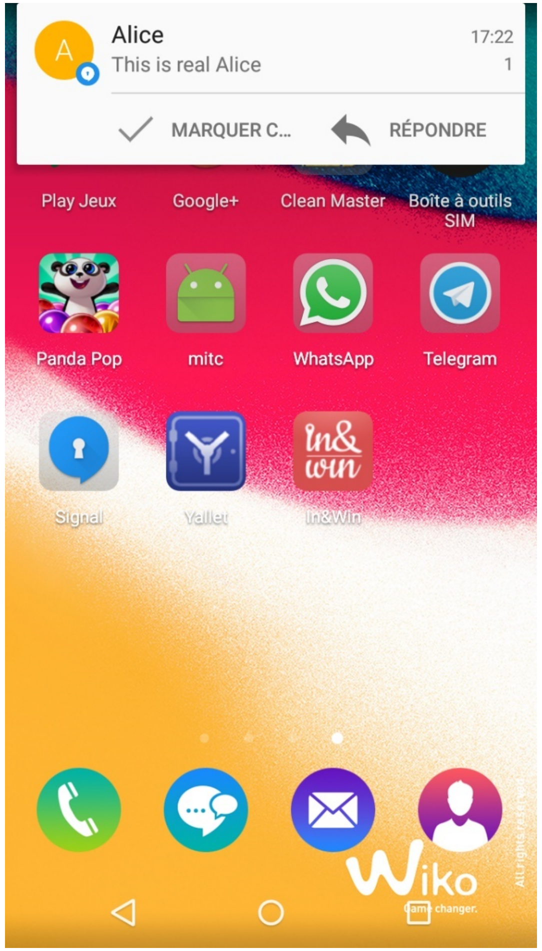
Alice 17:17
This is Eve as " Alice" 1



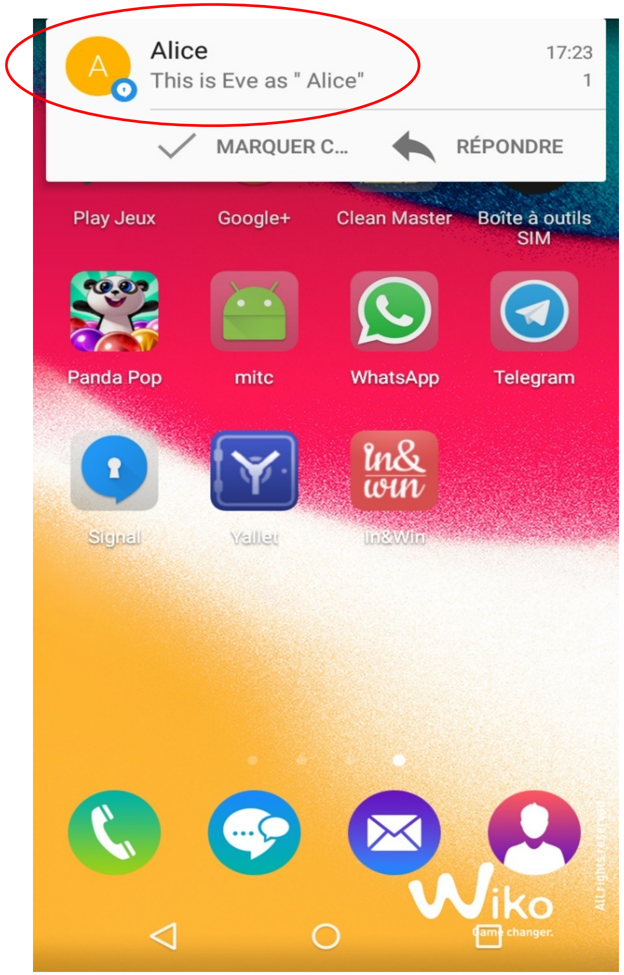
Alice 18:03
This is Eve as " Alice"

Alice 18:02
This is real Alice

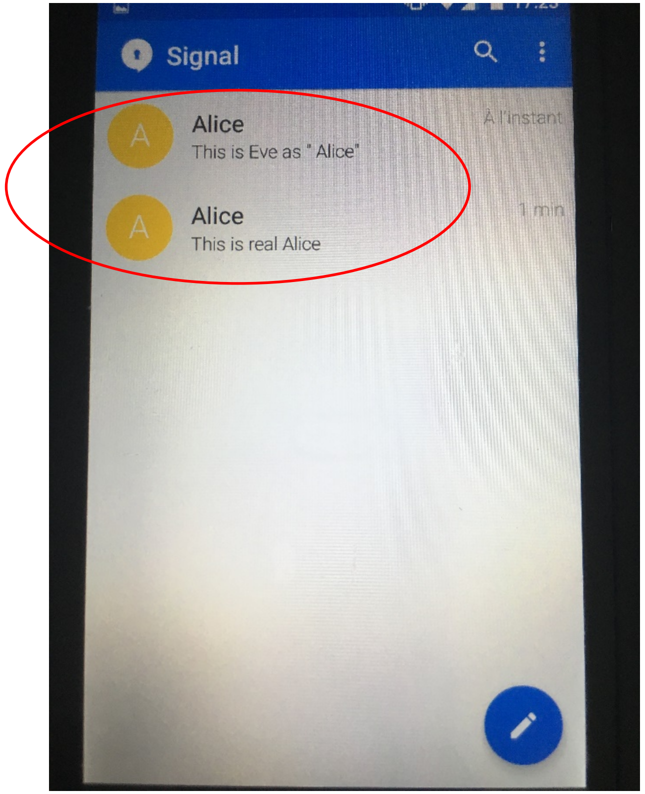
Let’s create a new contact « Alice»



Alice 17:22
This is real Alice 1



Alice 17:23
This is Eve as " Alice" 1



Alice À l'instant
This is Eve as " Alice"

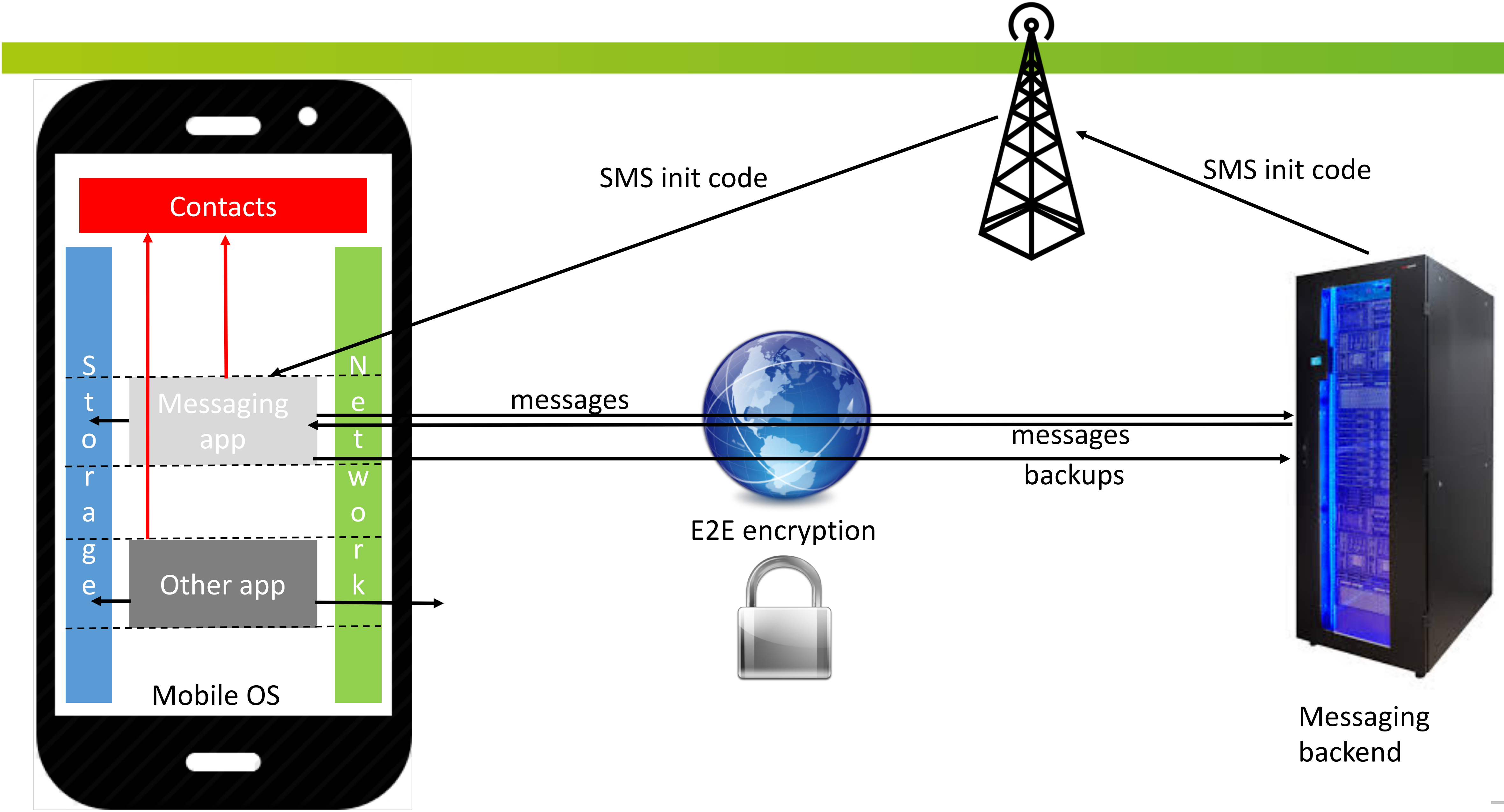
Alice 1 min
This is real Alice

Why does it work ?

- **Design error** from a security point of view:
  phone number as implicit identifier is a poor choice
- **Abusing Trust On First Use** (TOFU):
  new contact = new key = accepted by default
- Same old trick of invisible characters

- **End user/mobile not really included in the threat model**
  - Focus on protecting network/servers (e.g from government agencies)
  - Side channel attack with some social engineering out of scope
  - Formal security analysis of Signal protocol: https://eprint.iacr.org/2016/1013.pdf
    Signal specifies a mandatory method for participants to verify each other's identity keys through an out-of-band channel, but most implementations do not require such verification to take place before messaging can occur

Threat model: mobile focus & simplified



What can we do with MITC ?

- **Man In The Middle**
  - Showed theoretical attack at DefCon Crypto village in 2016
  - Conversation is end-to-end encrypted but Alice is not talking to Bob directly: Eve pretends to be « Bob» and forwards messages as « Alice»

- **Spear phishing ultimate weapon**
  - Demonstration at OWASP AppSec EU in 2018 with Laureline DAVID
  - Android game: a social version of Rock, Paper, Scissors
    - Available on Google PlayStore at https://play.google.com/store/apps/details?id=com.tricktrap.rps
    - Approved without any issue since July 2018
    - Public source code: https://github.com/ltouroumov/rockpaperspam-client
  - Command-and-Control server
    - **Web interface to send a malicious link pretending to come from a friend**
    - Public source code: https://github.com/ltouroumov/rockpaperspam-server

Risk assessment

- Simple evaluation: risk = easiness of attack * user impact
- Difficulty of attack: Low-Medium
  - Technically: Low
    - Easy to access contacts via code
    - Not a problem to get MITC application approved for publication
  - Logistics : Medium
    - One phone number is enough
    - Need to convince many users to install the MITC application
    - But « Ponzi scheme » possible by using the contact information
- Impact: High
  - Thousands of users can be targeted: multi-app

Difficulty to attack	Low business impact	Medium business impact	High business impact
Low	Low	Medium	Very High
Medium	Low	Medium	High
High	Low	Low	Medium

Vendors feedback

- Telegram: security@telegram.org = **/dev/null**

- WhatsApp (Facebook)

  We appreciate your report. ***Ultimately*** *an attacker with* ***malware*** *installed on a device is going to be able to alter data on the device itself. In your examples for* ***WhatsApp conversations remained properly bound to the phone number that the messages were sent to****. Beyond that, WhatsApp allows people to* ***set local aliases for contacts*** *and to view the number associated with a specific message thread at any point. Given that, we don't feel that this behavior poses a significant risk and* ***we do not plan to make any changes here****. Please* ***let us know if you feel we've misunderstood something*** *here!*

- Signal (Moxie Marlinkspike)

  Hey Jeremy, saw your support email about "man in the contacts." This, like all interception techniques, is ***what safety numbers are for****.* ***Signal users would be notified that*** *the safety numbers for* ***their contact have changed****, and be asked to verify them. A successful MITM attack would need to find a way to intercept communication without triggering that notice.*

  Hey Jeremy, ***Signal is not designed to protect your device against malware****.*
  Thanks for getting in touch, good luck with everything.

Countermeasures: wait for fixes ?

- Mobile OS
  - Sandbox contact information
  - Be stricter on write operation to address book

- Secure messengers
  - Give up the implicit trust on contact information:
    require users to manually add people they are talking to
  - **Raise user awareness when a conversation is starting with a brand new contact:**
    make it clear in UI this is an unusual situation, e.g. with a danger sign

Countermeasures: your company

- **Leverage your MDM** for corporate devices
  - Whitelist applications that can be installed:
    this will limit the risk of tampering the address book
  - Study if possible to overwrite address book with corporate directory info
- For personal devices, **train users** to be careful with brand new conversations
  - Don’t reply directly from notification, have a look at the history before
- Use Threema corporate version
  - Swiss German app
  - Manual id handling, with optional contact sync
  - Visible trust level: Red/Orange/Green
  - Questions on contacts handling sent to press@threema.ch
    Very detailed answer with the clear design choices received the next day

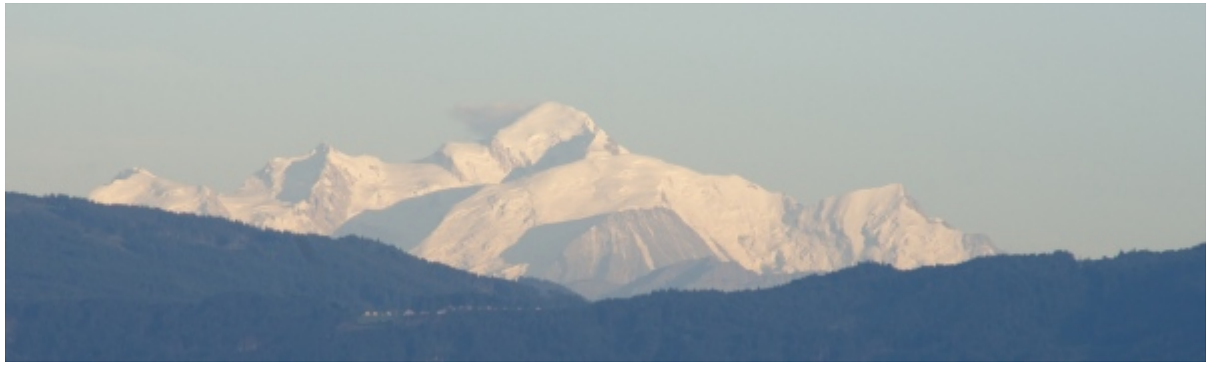
Conclusion

- **E2E can't bring trust if you're not sure who you're talking to**
  - The great security reputation of those messegners can be used against your organization for a successful social engineering attack

- **Security model around contacts is far too open** for sensitive apps
  - Having control on the content of the address book for corporate devices is absolutely necessary

- **Do have a look at the conversation history**, rather than interacting directly within the push notification
  - when writing an answer
  - before clicking on a link:
    E2E is by design blind to malicious content

Thank you !



Any question ?

contact@securingapps.com