Secure and Usable Mobile Solutions for Authentication and Single Sign-On: a Methodology for their Design and Assessment

Roberto Carbone - Silvio Ranise - Giada Sciarretta

https://st.fbk.eu/workshop-ifipsc-18
Question 1: Mobile vs Browser-based Authentication

Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen

Question 2: e-health legal compliance

Exercise 3: TreC activation phase

Question 3: Pros & Cons of our methodology and TreC solution

Question 1bis: Mobile vs Browser-based Authentication
● Which is your background?

[ ] IT Security
[ ] Legal
[ ] Other

● Which is your position?

[ ] Master Student
[ ] PhD Student
[ ] Researcher
[ ] Other
Outline

- Introduction and Problem Statement
  - Question 1: Mobile vs Browser-based Authentication

- Design Choices: Security and Usability Problems
  - Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen

- Methodology Overview: TreC Scenario
  - Question 2: e-health legal compliance

- Usability Discussion on TreC
  - Exercise 3: TreC activation phase

- Conclusions and On-going/Future Work
  - Question 3: Pros & Cons of our methodology and TreC solution
  - Question 1bis: Mobile vs Browser-based Authentication
Digital Identities

- We use our **digital identities** everyday, from accessing social apps to security-critical apps.
Password-based authentication is no longer sufficient in terms of security.

54% of people use 5 or fewer passwords across their entire online life[1]

Password-based authentication is no longer sufficient in terms of security.

54% of people use 5 or fewer passwords across their entire online life.[1]

Single Sign-On (SSO) allows users to access multiple apps through a single authentication act.

**SAML 2.0**

consolidated, corporate & governmental environments

used for social network (billions of users)
Psw-based Authn and Browser-based SSO Protocol

Authorization Code Flow

Browser: SP Login Page
- **Auth Request**, `client_id + redirect_uri`

SP backend
- **User Login + Consent**, `code`

OIDC backend
- **Token Request**, `code + client_id + client_secret`
- **access_token + id_token**
Psw-based Authn and Browser-based SSO Protocol

Authorization Code Flow

Browser: **OIDC** Login Page

- **UA**
- Auth Request, client_id + redirect_uri
- User Login + Consent
  - code

SP backend

- Token Request, code + client_id + client_secret
  - access_token + id_token

OIDC backend
Authorization Code Flow

Browser: **SP Home Page**

- **SP backend**
  - **Auth Request, client_id + redirect_uri**
  - **User Login + Consent**
  - **code**

- **OIDC backend**
  - **Token Request, code + client_id + client_secret**
  - **access_token + id_token**
Single Sign-On (SSO) allows users to access multiple apps through a single authentication act.

- **Usability**: only a password to remember for several apps
- **Security**: more complex passwords
- **Usability**: shared sessions
- **Security**: Only 1 password to compromise

**SSO + Multi-Factor Authentication solutions**
Multi-Factor Authentication (MFA)

A procedure based on the use of two or more of the following factors:

- **knowledge**, something only the user knows, e.g., static password, personal identification number;
- **ownership**, something only the user possesses, e.g., token, smart card, mobile phone; and
- **inherence**, something the user is, e.g., biometric characteristic, such as a fingerprint.

MFA procedure requires the generation of a **One Time Password (OTP)**

- is an una tantum code with a short expiration
- proofs the possession of the OTP generator (hardware, mobile app...) and/or of the device that received it, and
- [optionally] proofs the knowledge of the PIN used to activate the OTP generator

### Time-based OTP (TOTP)

1. **Seed** → **OTP** = 472894
2. **Seed** → **OTP** = 472894
3. **OTP** = 472894
4. **Server**

### Challenge-Response

1. **Challenge**
2. **OTP** = **Response**
3. **User Device**
4. **Server**
Many MFA Solutions on the Market

“FIDO is the World’s Largest Ecosystem for Standards-Based, Interoperable Authentication”

Allows **online services** to augment the security of their existing password infrastructure by adding a **strong second factor** to user login

User needs a **FIDO U2F device**
Main Focus of this Workshop

Design and Security Assessment of SSO and MFA Solutions for Mobile Native Applications

ACME OTP

OTP value

47820512

44 seconds left

New OTP
Mobile Native Apps vs Browser-based Apps

Native Apps

Browser-based Apps
Mobile Native Apps vs Browser-based Apps

Native Apps

Browser-based Apps

Read reviews on web. Want to write one? Use the app
Can we use browser-based authentication and SSO solutions for mobile apps?
NO. Browser-based protocol usages on mobile applications require **detailed understandings** about the protocol specifications and the mobile platform capabilities → a new design is needed.
Mobile App vs Browser-based App

**NO.** Browser-based protocol usages on mobile applications require **detailed understandings** about the protocol specifications and the mobile platform capabilities → **a new design is needed**

---

---


NO. Browser-based protocol usages on mobile applications require **detailed understandings** about the protocol specifications and the mobile platform capabilities → a new design is needed


NO. Browser-based protocol usages on mobile applications require detailed understandings about the protocol specifications and the mobile platform capabilities → a new design is needed

Different redirection mechanisms

- Auth Request, client_id + redirect_uri
- User Login + Consent
- code
- Token Request, code + client_id + client_secret
- access_token + id_token


Mobile App vs Browser-based App

NO. Browser-based protocol usages on mobile applications require detailed understandings about the protocol specifications and the mobile platform capabilities → a new design is needed

Different redirection mechanisms

Wrong design choices could lead to security and usability problems

Token Request, code + client_id + client_secret

access_token + id_token


IdM Protocols for Mobile App

- Proprietary Solutions

- OAuth/OIDC Working Group has released guidelines to support Single Sign-On for mobile native apps
  - **OpenID Connect Native Application Token Agent Core 1.0 (NAPPS) (2015)** - ONLY a DRAFT (now abandoned)
  - **OAuth for native apps [RFC 8252] (2017)** - BEST CURRENT PRACTICE
IdM Protocols for Mobile App: limitations

- Proprietary Solutions

- OAuth/OIDC Working Group has released guidelines to support Single Sign-On for mobile native apps
  - OpenID Connect Native Application Token Agent Core 1.0 (NAPPS) (2015) - ONLY a DRAFT (now abandoned)
  - OAuth for native apps [RFC 8252] (2017) - BEST CURRENT PRACTICE

Technical limitations: non-obvious support to SAML and MFA in native mobile apps

(2017) - BEST CURRENT PRACTICE
Outline

- Introduction and Problem Statement
  - Question 1: Mobile vs Browser-based Authentication

- Design Choices: Security and Usability Problems
  - Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen

- Methodology Overview: TreC Scenario
  - Question 2: e-health legal compliance

- Usability Discussion on TreC
  - Exercise 3: TreC activation phase

- Conclusions and On-going/Future Work
  - Question 3: Pros & Cons of our methodology and TreC solution
  - Question 1bis: Mobile vs Browser-based Authentication
Design Choices

TODO List

1. design of an IdM Solution

2. security analysis
Design Choices

TODO List

1. design of an IdM Solution
   - How to establish trust?
   - How do communication channels work?

2. security analysis
Design Choices

TODO List

1. design of an IdM Solution
   - How to establish trust?
   - How do communication channels work?
   - Legal aspect?!?

2. security analysis
Design Choices

**TODO List**

1. design of an IdM Solution
   - How to establish trust?
     - How do communication channels work?
   - Legal aspect?!?

2. security analysis
   - Which tool? How to formalize the security properties?
Wrong design choices could lead to security and usability problems
Example of wrong design

2FA Choice: SMS
Example of wrong design

2FA Choice: SMS

Table 8-1 Authenticator Threats

<table>
<thead>
<tr>
<th>Authenticator Threat/Attack</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Social Engineering</td>
<td>The attacker establishes a level of trust with a subscriber in order to convince the subscriber to reveal their authenticator secret or authenticator output.</td>
<td>An out of band secret sent via SMS is received by an attacker who has convinced the mobile operator to redirect the victim’s mobile phone to the attacker.</td>
</tr>
<tr>
<td>Endpoint Compromise</td>
<td>Malicious code on the endpoint causes authentication to other than the intended verifier.</td>
<td>A malicious app on the endpoint reads an out-of-band secret sent via SMS and the attacker uses the secret to authenticate.</td>
</tr>
</tbody>
</table>
Example of wrong design

We had a security incident. Here's what you need to know.

**TL;DR:** A hacker broke into a few of Reddit's systems and managed to access some user data, including some current email addresses and a 2007 database backup containing old salted and hashed passwords. Since then we've been conducting a painstaking investigation to figure out just what was accessed, and to improve our systems and processes to prevent this from happening again.

**What happened?**

On June 19, we learned that between June 14 and June 18, an attacker compromized a few of our employees' accounts with our cloud and source code hosting providers. Already having our primary access points for code and infrastructure behind strong authentication requiring two factor authentication (2FA), we learned that SMS-based authentication is not nearly as secure as we would hope, and the main attack was via SMS intercept. We point this out to encourage everyone here to move to token-based 2FA.

Although this was a serious attack, the attacker did not gain write access to Reddit systems; they gained read-only access to some systems that contained backup data, source code and other logs. They were not able to alter Reddit information, and we have taken steps since the event to further lock down and rotate all production secrets and API keys, and to enhance our logging and monitoring systems.

Now that we've concluded our investigation sufficiently to understand the impact, we want to share what we know, how it may impact you, and what we've done to protect us and you from this kind of attack in the future.

**What information was involved?**

Since June 19, we've been working with cloud and source code hosting providers to get the best possible understanding of what data the attacker accessed. We want you to know about two key areas of user data that was accessed:

- **All Reddit data from 2007 and before including account credentials and email addresses**
  - *What was accessed:* A complete copy of an old database backup containing very early Reddit user data — from the site's launch in 2005 through May 2007. In Reddit's first years it had many fewer features, so the most significant data contained in this backup was the profile pages (user names, passwords, email addresses) and account creation dates.
SIM hijacking attacks can result in account compromises by stealing one-time passcodes sent over SMS.

A cryptocurrency investor is suing AT&T for $224 million, alleging he lost $24 million in virtual currency after the carrier failed to stop two separate attacks where his phone number was commandeered by attackers.

See Also: Preventing an Inside Job: Detection, Technology and People

Michael Terpin, who runs an investment group called Bit Angels and is involved in the cryptocurrency community, is seeking $24 million in compensatory damages and $200 million in punitive damages, according to the lawsuit, which was filed on Wednesday in federal court in Los Angeles.

Terpin was a victim of two SIM hijacking attacks, which are sometimes referred to as SIM swapping or port-out scams. The attack involves an attacker convincing a mobile provider to move a number to a different SIM card. Many times, the targets of such attackers are those with large holdings of bitcoin and other cryptocurrencies (see Cryptocurrency Theft: $1.1 Billion Stolen in Last 6 Months).

Attackers can successfully take over someone’s phone number by tricking an employee at a carrier that they’re the legitimate account holder. In other cases, telecom employees may be crooked and actually be the ones pulling the attacks.
Exercise 1

User Agent (UA) Choice: embedded browser

1. How does an embedded browser work?
2. Are there any security issues?
3. What happens to the user experience when accessing multiple apps?
Exercise 2

OTP Choice: app that shows the OTP value

1. What happens to the user experience when accessing a mobile native app?
2. Are there any security issues?
3. List one or more OTP choice alternatives
Exercise 1 (UA Choice): embedded browser

1. How does an embedded browser work?
2. Are there any security issues?
3. What happens to the user experience when accessing multiple apps?

Exercise 2 (OTP Choice): app that shows the OTP value

1. What happens to the user experience when accessing a mobile native app?
2. Are there any security issues?
3. List one or more OTP choice alternatives
Exercise 1: Answer 1 - definition

User Agent (UA) Choice: embedded browser

“a user-agent hosted inside the native app itself (such as via a web-view), with which the app has control over to the extent it is capable of accessing the cookie storage and/or modifying the page content”

Exercise 1: Answer 2 - security

User Agent (UA) Choice: embedded browser

Security

Impact: the attacker can access other SP apps as the user

SP4 News adds some javascript to read user’s credentials

```
webView.evaluateJavascript(
    "(function() { return 
    document.getElementById('password').value; })();",
    new ValueCallBack<String>() {
        @Override public void onReceiveValue(String s)
        {
            Log.d("WebViewField",s);
        }
    });
```

Exercise 1: Answer 3 - usability

User Agent (UA) Choice: embedded browser

**Security**

**Impact:** the attacker can access other SP apps as the user

**Usability:** no SSO


Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

![ACME Calendar app with a login button](image)

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

![Image of a smartphone with a login screen for ACME IdP]

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
**Exercise 2: Answer 1 - usability**

**OTP Choice:** app that shows the OTP value

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

**OTP Choice:** app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 1 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
**Exercise 2: Answer 1 - usability**

**OTP Choice:** app that shows the OTP value

![OTP display](https://ACMEIdP.it/login)

**Usability:** move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 2 - usability

OTP Choice: app that shows the OTP value

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 2 - security

OTP Choice: app that shows the OTP value

IdP of SPID

ACME Calendar

Hi Alice!

February 2018

<table>
<thead>
<tr>
<th>M</th>
<th>T</th>
<th>W</th>
<th>T</th>
<th>F</th>
<th>S</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>30</td>
<td>31</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
</tr>
<tr>
<td>27</td>
<td>28</td>
<td>29</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

Security: Copy&Paste

Usability: move from an app to another (burdensome for the user in terms of time and difficulty)
Exercise 2: Answer 3 - alternatives

OTP Choice: app that shows the OTP value

Alternatives:

- OTP app that does not ask the user to enter the OTP; after the PIN input, the OTP value is sent to the IdP in a transparent way

- Use of external OTP generators

Security Card

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>794</td>
<td>536</td>
<td>495</td>
<td>096</td>
<td>555</td>
<td>748</td>
<td>904</td>
<td>565</td>
<td>125</td>
<td>401</td>
</tr>
<tr>
<td>11</td>
<td>295</td>
<td>325</td>
<td>137</td>
<td>107</td>
<td>404</td>
<td>820</td>
<td>040</td>
<td>599</td>
<td>605</td>
<td>013</td>
</tr>
<tr>
<td>21</td>
<td>839</td>
<td>558</td>
<td>530</td>
<td>171</td>
<td>012</td>
<td>399</td>
<td>510</td>
<td>024</td>
<td>089</td>
<td>548</td>
</tr>
<tr>
<td>31</td>
<td>106</td>
<td>486</td>
<td>777</td>
<td>102</td>
<td>962</td>
<td>769</td>
<td>825</td>
<td>421</td>
<td>455</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>727</td>
<td>729</td>
<td>444</td>
<td>632</td>
<td>016</td>
<td>526</td>
<td>009</td>
<td>490</td>
<td>549</td>
<td>943</td>
</tr>
</tbody>
</table>
Outline

• Introduction and Problem Statement
  
  Question 1: Mobile vs Browser-based Authentication

• Design Choices: Security and Usability Problems
  
  Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen

• Methodology Overview: TreC Scenario
  
  Question 2: e-health legal compliance

• Usability Discussion on TreC
  
  Exercise 3: TreC activation phase

• Conclusions and On-going/Future Work
  
  Question 3: Pros & Cons of our methodology and TreC solution

Question 1bis: Mobile vs Browser-based Authentication
Design for an IdM Solution

We provide:

- a **reference model mID(OTP)** for mobile authentication and SSO solutions
- a **methodology** to assist the designer in the customization of mID(OTP) and in the analysis of its security and usability
Reference Model - mID(OTP)

- **mID(OTP)** is inspired to:
  - a rational reconstruction of Facebook solution (UA=app), and
  - an analysis of OAuth for native app (UA=network)

- **the OTP generation approaches**: Time-based OTP (TOTP) and Challenge-Response
Methodology Overview

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

Phase 1: AppCtx Identification
- AppCtx Table
  - Entities
  - Data Nature
  - AuthN Aspects
  - UA/OTP Choices

Phase 2: Customization of mID(OTP)
- MSC, Asms and Goals of mID(OTP)
- Customized mID(OTP)
  - MSC, Asms and Goals of scenario

Phase 3: Security Analysis
- Semiformal
  - Formal
- Security Analysis Report
- Did the analysis find attacks?
  - no
  - yes
- Did the analysis identify usability problems?
  - no
  - yes

Phase 4: Usability Analysis
- Questionnaires/ examples
- Usability Analysis Report

Designer
Methodology Overview

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

Phase 1: AppCtx Identification

Phase 2: Customization of mID(OTP)

Phase 3: Security Analysis

Phase 4: Usability Analysis

Risk Assessment Report

Security Analysis Report

Did the analysis find attacks?

Did the analysis identify usability problems?

Customized IdM solution

Designer
Methodology Overview

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

Phase 1: AppCtx Identification
- AppCtx Table
  - Entities
  - Data Nature
  - AuthN Aspects
  - UA/OTP Choices

Phase 2: Customization of mID(OTP)
- MSC, Asm and Goals of mID(OTP)
- Customized mID(OTP)
  - MSC, Asm and Goals of scenario

Phase 3: Security Analysis
- Semiformal / Formal
- Security Analysis Report
- Did the analysis find attacks?
  - no
  - yes

Phase 4: Usability Analysis
- Questionnaires
  - examples
- Usability Analysis Report
- Did the analysis identify usability problems?
  - yes
  - no
- Customized IdM solution
Methodology Overview

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

Phase 1: AppCtx Identification

Phase 2: Customization of mID(OTP)

Phase 3: Security Analysis

Phase 4: Usability Analysis

Risk Assessment Report

Security Analysis Report

Did the analysis find attacks?

Did the analysis identify usability problems?

Customized IdM solution
Real-World Scenarios

1. **TreC**: a multi-factor authentication solution with a single sign-on experience for mobile e-Health applications.

2. **Smart Community**: a secure delegated access solution in the context of smart-cities.

3. **FIDES**: an IdM solution that combines federation and cross-border aspects in the context of the European single digital market.

4. **DigiMat-Lab (Istituto Poligrafico e Zecca dello Stato)**: a mobile multi-factor authentication solution that uses the Italian electronic identity card (**CIE 3.0**) as second factor.
TreC Platform

TreC ("Cartella Clinica del Cittadino") is a Citizen-controlled PHR (Personal Health Record) connected to the national EHR (Fascicolo Sanitario Nazionale)

Goal of TreC: empowering citizens to manage their own health and facilitating communications between patients and healthcare professionals and facilities

Subscribers: 81,587
TreC: Web and Mobile apps

Self-management
Remote monitoring
TreC: Web and Mobile apps

- Self-management
- Remote monitoring
TreC: Web and Mobile apps

Goal: provide a multi-factor authentication solution and a SSO experience for the mobile apps of TreC
Phase 1: Fill AppCtx Table

| Entities          | User → Patient; SP<sub>app</sub> → TreC Referti; SP<sub>S</sub> → TreC; UA,TP<sub>app</sub> → OTP-PAT; IdP<sub>S</sub>, TP<sub>S</sub> → ADC; |
### Phase 1: Fill AppCtx Table

**Entities**

| Entities | User → **Patient**; 
|          | SP<sub>app</sub> → **TreC Referti**; SP<sub>S</sub> → **TreC**; 
|          | UA, TP<sub>app</sub> → **OTP-PAT**; IdP<sub>S</sub>, TP<sub>S</sub> → **ADC**; 

**UA choice**

- [ ] Browser
- [✓] Application

---

**Diagram:**

- **User**
- **Patient**
- **TreC Referti**
- **TreC Server**
- **ADC**
- **IdP+TP**
- **SP**

**Note:** The diagram illustrates the flow of entities and interactions, with User choosing between Browser and Application as their UA choice.
<table>
<thead>
<tr>
<th>Data Nature</th>
<th>anonymous</th>
<th>personal</th>
<th>sensitive</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthN Aspects</td>
<td>MFA support?</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Session handling?</td>
<td>yes</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>OTP choice</td>
<td>TOTP</td>
<td>CR</td>
<td>other</td>
</tr>
</tbody>
</table>

Patient

OTP-PAT

TreC Referti

ADC

TreC Server


“data controllers shall be required to adopt the minimum security measures in order to ensure a minimum level of personal data protection”

In case of public administration, the CAD (Codice dell'Amministrazione Digitale - D.Lgs.n. 82/3005) must be followed.


Italian national eID scheme (SPID)

SPID 2 (LoA3 of ISO-IEC 29115) for PHR

https://www.spid.gov.it/
Which legal obligations do you have to follow when dealing with e-health data in your country?
## AppCtx Table - TreC

<table>
<thead>
<tr>
<th>Entities</th>
<th>User → Patient; SP_{app} → TreC_Refer; SP_{S} → TreC; UA,TP_{app} → OTP-PAT; IdP_{S},TP_{S} → ADC;</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA choice</td>
<td>□ Browser ✓ Application</td>
</tr>
<tr>
<td>Data Nature</td>
<td>□ anonymous ✓ personal ✓ sensitive</td>
</tr>
<tr>
<td>AuthN Aspects</td>
<td>MFA support? ✓ yes □ no; Session handling? ✓ yes □ no</td>
</tr>
<tr>
<td>OTP choice</td>
<td>✓ TOTP □ CR □ other</td>
</tr>
</tbody>
</table>
Phase 2: Customization

AppCtx Table - TreC

<table>
<thead>
<tr>
<th>Entities</th>
<th>User → Patient; SP → TreC Refert; SP → TreC; UA, TP → OTP-PAT; IdP, TP → ADC;</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA choice</td>
<td>Browser [x] Application [✓]</td>
</tr>
<tr>
<td>Data Nature</td>
<td>anonymous [✓] personal [x] sensitive [✓]</td>
</tr>
<tr>
<td>AuthN Aspects</td>
<td>MFA support? [✓] yes [✓] no</td>
</tr>
<tr>
<td>OTP choice</td>
<td>TOTP [✓] CR [✓] OTHER [✓]</td>
</tr>
</tbody>
</table>

Phase 2: Customization of mID(OTP)

- Is UA=app? no
- MFA? no
- Security Goals
  - G1BA: SP auth User
  - G1MFA: SP auth User with MFA, and G2: OTP proves its origin and is fresh
- OTP box
  - N.A.
  - TOTP: TreC Scenario
  - CR: CIE Scenario

Customized mID(OTP)

MSC Scenario

Goals Scenario

Asms Scenario

AdC [✓] TreC [✓] Pat. [✓]
Phase 2: Customization

AppCtx Table - TreC

<table>
<thead>
<tr>
<th>Entities</th>
<th>User → Patient; SP → TreC Refert.; SP → TreC; UA; TP → OTP; IdP; TP → ADC;</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA choice</td>
<td>Browser ☐ Application ☑</td>
</tr>
<tr>
<td>Data Nature</td>
<td>anonymous ☐ personal ☑ sensitive ☐</td>
</tr>
<tr>
<td>AuthN Aspects</td>
<td>MFA support? ☑ yes ☐ no. Session handling? ☑ yes ☐ no</td>
</tr>
<tr>
<td>OTP choice</td>
<td>TOTP ☑ CR ☐ other</td>
</tr>
</tbody>
</table>

MSC

Security Assumptions (Asms)
- TA, BA1, BA2, CA1, CA2, CA3, AA, UBA1, UBA2
- TA, BA1, BA2, CA1-A2', CA3', AA, UBA1', UBA2'

Security Goals
- G1BA: SP auth User
- G1MFA: SP auth User with MFA, and
- G2: OTP proves its origin and is fresh

MFA?

Is UA=app?

Yes

OTP box
- N.A.
- TOTP: TreC Scenario
- CR: CIE Scenario

No

Goals Scenario

Asms Scenario

Customized mID(OTP)

MSC Scenario
mID(OTP) requires 3 phases:

Registration: is performed by the TreC developer to register the app with ADC. It is performed just once.

Activation: is performed by the Patient to configure OTP-PAT. It is performed the first time only.

Exploitation: is performed every time Patient accesses TreC.
GOAL: registration of TreC with ADC

TreC dev has to provide some information, such as the app package name and the certificate fingerprint (**key_hash**) of the app.

**key_hash** is a digest of the le CERT.RSA, that contains the public key of the developer, the signature of the app package (APK) obtained with the private key of the developer and other information about the certificate.
GOAL: enable OTP-PAT to securely interact with ADC.

1. Laptop
   Using a portal made available by ADC, User logs in with CPS and obtains an activation_code.
GOAL: enable OTP-PAT to securely interact with ADC.

1. Laptop
   Using a portal made available by ADC, User logs in with CPS and obtains an activation_code.

2. Mobile
   On her mobile, User enters the activation_code into OTP-PAT and generates her PIN
**GOAL**: user logs in TreC app using the ADC identity
GOAL: user logs in TreC app using the ADC identity
GOAL: user logs in TreC app using the ADC identity
**GOAL**: user logs in TreC app using the ADC identity
**GOAL**: user logs in TreC app using the ADC identity
GOAL: user logs in TreC app using the ADC identity
GOAL: user logs in TreC app using the ADC identity
GOAL: user logs in TreC app using the ADC identity
Goals

AppCtx Table - TreC
Phase 2: Customization

The TreC solution is a 3 instance-factors authentication solution:

1. `token_IdP` that is stored in OTP-PAT and in ADC as a result of the activation phase (used as a session token in place of the user credentials to provide a SSO experience);

2. PIN known by Patient to unlock OTP-PAT;

3. `{seed}_PIN` that is stored in OTP-PAT.

MFA Goal:
TreC authenticates Patient even if an intruder knows up to 2 instance-factors

G. Sciarretta, R. Carbone, S. Ranise and L. Viganò. 
Phase 2: Customization

AppCtx Table - TreC

<table>
<thead>
<tr>
<th>Entities</th>
<th>User → Patient; SP → TreC Referent; SP → TreC; UA, TP → OTP-PAT; IdP, TP → ADC;</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA choice</td>
<td>Browser ☐ Application ☑</td>
</tr>
<tr>
<td>Data Nature</td>
<td>anonymous ☐ personal ☑ sensitive ☑</td>
</tr>
<tr>
<td>AuthN</td>
<td>MFA support? ☑ yes ☐ no. Session handling? ☑ yes ☐ no</td>
</tr>
<tr>
<td>OTP choice</td>
<td>TOTP ☑ CR ☐ other □</td>
</tr>
</tbody>
</table>

Phase 2: Customization of mID(OTP)

- Is UA=app? yes → MSC
- Security Assumptions (Asms)
  - TA, BA1, BA2, CA1, CA2, CA3, AA, UBA1, UBA2

- Is UA=app? no → MFA?
- MFA? no → MSC
- Security Assumptions (Asms)
  - TA, BA1, BA2, CA1-A2', CA3', AA, UBA1', UBA2'
- MFA? yes → MSC
- Security Goals
  - G1BA: SP auth User
  - G2: OTP proves its origin and is fresh

Customized mID(OTP)

- MSC Scenario
- Goals Scenario
- Asms Scenario

ADC is trusted by TreC on identity assertions
BA1 Integrity and confidentiality of data stored in the device
...
## Assumptions

### Strong Assumptions

<table>
<thead>
<tr>
<th>Trust Assumption</th>
<th>TA</th>
<th>ADC is trusted by TreC on identity assertions.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Assumptions</td>
<td>CA1</td>
<td>The communication between TreC and OTP-PAT is carried over an inter-app communication implemented using StartActivityForResult(). This Android method --- which allows an app to execute another app and get a result back --- guarantees that TreC that sends a request to OTP-PAT at Step A2 in Figure 6.1 is the same app that receives the result back from OTP-PAT at Step A10.</td>
</tr>
<tr>
<td></td>
<td>CA2</td>
<td>To read the key hash value (Step A3 of Figure 6.1), OTP-PAT uses the Android method getPackageInfo(client packageName, PackageManager.GET SIGNATURES), which extracts the information about the certificate fingerprint included in the package of TreC.</td>
</tr>
<tr>
<td></td>
<td>CA3</td>
<td>The communication between OTP-PAT and ADC occurs over a unilateral SSL or TLS channel (henceforth SSL/TLS), established through the exchange of a valid certificate (from ADC to OTP-PAT).</td>
</tr>
<tr>
<td>Activation Assumption</td>
<td>AA</td>
<td>The activation phase is correctly performed by Patient. That is, Patient downloads the correct OTP-PAT (i.e. it is not fake app) and correctly follows the activation phase process, and the communication channels that are involved in this phase are secure.</td>
</tr>
</tbody>
</table>

### Weak Assumptions

<table>
<thead>
<tr>
<th>Background Assumptions</th>
<th>BA1</th>
<th>Integrity and confidentiality of data stored in the device, i.e. an app cannot read or modify data stored by another app.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BA2</td>
<td>There is no surveillance software (e.g., keylogger) installed on the user's device capable of reading the values that Patient types.</td>
</tr>
<tr>
<td>User Behaviour Assumptions</td>
<td>UBA1</td>
<td>Patient enters her credentials and (optionally) values for the OTP generation only in the correct OTP-PAT app being careful not to be seen by other people.</td>
</tr>
<tr>
<td></td>
<td>UBA2</td>
<td>Patient is the only person using the OTP-PAT app that has been activated with her identity.</td>
</tr>
</tbody>
</table>
### Assumptions

#### Strong Assumptions

<table>
<thead>
<tr>
<th>Trust Assumption</th>
<th>TA</th>
<th>ADC is trusted by TreC on identity assertions.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Assumptions</td>
<td>CA1</td>
<td>The communication between TreC and OTP-PAT is carried over an inter-app communication implemented using StartActivityResult(). This Android method --- which allows an app to execute another app and get a result back --- guarantees that TreC that sends a request to OTP-PAT at Step A2 in Figure 6.1 is the same app that receives the result back from OTP-PAT at Step A10.</td>
</tr>
<tr>
<td>CA2</td>
<td>To read the key hash value (Step A3 of Figure 6.1), OTP-PAT uses the Android method getPackageInfo(client packageName, PackageManager.GET_SIGNATURES), which extracts the</td>
<td></td>
</tr>
<tr>
<td>CA3</td>
<td>The communication between OTP-PAT and ADC occurs over a unilateral SSL or TLS channel (henceforth SSL/TLS), established through the exchange of a valid certificate (from ADC to OTP-PAT).</td>
<td></td>
</tr>
<tr>
<td>Activation Assumption</td>
<td>it is not fake app) and correctly follows the activation phase process, and the communication channels that are involved in this phase are secure.</td>
<td></td>
</tr>
</tbody>
</table>

#### Weak Assumptions

<table>
<thead>
<tr>
<th>Background Assumptions</th>
<th>BA1</th>
<th>Integrity and confidentiality of data stored in the device, i.e. an app cannot read or modify data stored by another app.</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA2</td>
<td>There is no surveillance software (e.g., keylogger) installed on the user's device capable of reading the values that Patient types.</td>
<td></td>
</tr>
<tr>
<td>User Behaviour Assumptions</td>
<td>UBA1</td>
<td>Patient enters her credentials and (optionally) values for the OTP generation only in the correct OTP-PAT app</td>
</tr>
<tr>
<td>UBA2</td>
<td>Patient is the only person using the OTP-PAT app that has been activated with her identity.</td>
<td></td>
</tr>
</tbody>
</table>
Phase 3: Security Analysis

Modelling the Honest Entities and the Intruder

```sql
entity ServiceProvider(Actor, OTPPAT, ADC, Patient: agent, Ch_T2O, Ch_02T, Ch_P2T, Ch_T2P: channel, Request: text) {
  select {on(Patient – Ch_P2T-> Actor: Request):{%STEP A1
    Actor – Ch_T2O-> OTPPAT: Actor; %STEP A2
    select {on(OTPPAT– Ch_02T->Actor:{ADC.?Patient.Actor}_inv(pk(ADC))):{%STEP A9
      SP_authn_U_on_Request: (Request) := Request;
    }
  }}
}

entity User_Agent(Actor, TreC, ADC, Patient: agent, Ch_T20, Ch_O2T, Ch_02A, Ch_A20, Ch_P20, Ch_02P: channel, Seed: seed, Token_IDP: token, CTime: time) {
  symbols
  KeyHash: key_hash;
  Metadata: text;
  PINRequest: text;
  PIN: pin;
  body{%of UA
  knows(CTime);
```
Modelling the Honest Entities and the Intruder

entity Service_Provider(Actor, OTP-PAT, ADC, Patient: agent, Ch_T20, Ch_02T, Ch_P2T, Ch_T2P: channel, Request: text) {  
  body(%of SP  
  selection(Patient → Ch_P2T → Actor: Request){ %STEP A1  
    Actor → Ch_T20 → OTP-PAT: Actor){ %STEP A2
      SP_authn_U_on_Request((Request) := Request;
      }
    }
  })
}

entity User_Agent(Actor, TreC, ADC, Patient: agent, Ch_T20, Ch_02T, Ch_02A, Ch_A20, Ch_P20, Ch_02P: channel, Seed: seed, Token_IDP: token, CTime: time) {
  symbols
  KeyHash: key_hash;
  Metadata: text;
  PINRequest: text;
  PIN: pin;

  body(%of UA  
  knows(CTime);
  
}
• **Dolev-Yao intruder** who can overhear and modify messages using his initial knowledge and the knowledge obtained from the traffic, but cryptography is secure, i.e. decryption is impossible without appropriate keys.

## Assumptions and Goal Formal Mapping

<table>
<thead>
<tr>
<th>Asm</th>
<th>Formal Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Specification of Assumptions</strong></td>
<td></td>
</tr>
<tr>
<td><strong>TA</strong></td>
<td>We do not consider sessions with i playing the role of ADC</td>
</tr>
<tr>
<td><strong>BA1</strong></td>
<td>&quot;Built-in&quot;: i cannot read the internal state of the other entities</td>
</tr>
<tr>
<td><strong>BA2</strong></td>
<td>&quot;Built-in&quot;: i cannot read the internal state of the other entities</td>
</tr>
<tr>
<td><strong>CA1</strong></td>
<td>link(T2O,O2T);</td>
</tr>
<tr>
<td><strong>CA2</strong></td>
<td>authentic_on(T2O,TreC);</td>
</tr>
<tr>
<td><strong>CA3</strong></td>
<td>confidential_to(O2A, ADC); weakly_authentic(O2A); weakly_confidential(A2O); authentic_on(A2O,ADC); link(O2A,A2O);</td>
</tr>
<tr>
<td><strong>AA</strong></td>
<td>Data obtained during the activation phase are nonpublic values</td>
</tr>
<tr>
<td><strong>UBA1</strong></td>
<td>confidential_to(P2O,OTFPAT);</td>
</tr>
<tr>
<td><strong>UBA2</strong></td>
<td>authentic_on(P2O,Patient);</td>
</tr>
</tbody>
</table>

\[ G_{1A} \quad SP\_authn\_U\_on\_Request:( ) Patient *->> TreC; \]
### Assumptions Formal Mapping

<table>
<thead>
<tr>
<th>Asm</th>
<th>Formal Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TA</strong></td>
<td>We do not consider sessions with ( i ) playing the role of ADC</td>
</tr>
<tr>
<td><strong>BA1</strong></td>
<td>&quot;Built-in&quot;: ( i ) cannot read the internal state of the other entities</td>
</tr>
<tr>
<td><strong>BA2</strong></td>
<td>&quot;Built-in&quot;: ( i ) cannot read the internal state of the other entities</td>
</tr>
<tr>
<td><strong>CA1</strong></td>
<td>link(T2O,O2T); DELETE link(T2O,O2T);</td>
</tr>
<tr>
<td><strong>CA2</strong></td>
<td>authentic_on(T2O,TreC); DELETE authentic_on(T2O,TreC);</td>
</tr>
<tr>
<td><strong>CA3</strong></td>
<td>confidential_to(O2A, ADC); weakly_authentic(O2A); weakly_confidential(A2O); authentic_on(A2O,ADC); link(O2A,A2O); DELETE confidential_to(O2A, ADC); weakly_authentic(O2A); weakly_confidential(A2O); authentic_on(A2O,ADC); link(O2A,A2O);</td>
</tr>
<tr>
<td><strong>AA</strong></td>
<td>Data obtained during the activation phase are nonpublic values</td>
</tr>
<tr>
<td><strong>UBA1</strong></td>
<td>confidential_to(P2O,OTPPAT); DELETE confidential_to(P2O,OTPPAT);</td>
</tr>
<tr>
<td><strong>UBA2</strong></td>
<td>authentic_on(P2O,Patient); DELETE authentic_on(P2O,Patient);</td>
</tr>
</tbody>
</table>

**G1_A**

\[ \text{SP_authn_U_on_Request:()} \text{ Patient} \xrightarrow{-\Rightarrow} \text{ TreC}; \]
Phase 3: Security Analysis

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

MSC Scenario

Asms Scenario

Goals Scenario

SAT-based bounded model checker: $M_S \ || \ M_I \models G_{BA}$
Security Analyses

Analysis 1: Is the solution secure under all the strong and weak assumptions?

SATMC does not find any attack on the solution (i.e. the intruder is not able to impersonate the user)
Security Analyses

Analysis 1: Is the solution secure under all the strong and weak assumptions?

SATMC does not find any attack on the solution (i.e. the intruder is not able to impersonate the user)

Analysis 2: Which assumptions can be removed? (e.g., modeling a wrong implementation)

- none of the strong assumptions
- one weak assumption at a time
- a combination of weak assumptions such that all the instance factors are not compromised only if the intruder compromises all the instance factors he is able to impersonate the patient
**Security Analyses**

**Analysis 1:** Is the solution secure under **all the strong and weak assumptions?**

SATMC does not find any attack on the solution (i.e. the intruder is not able to impersonate the user)

**Analysis 2:** Which assumptions can be removed? (e.g., modeling a wrong implementation)

- none of the strong assumptions
- one weak assumption at a time
- a combination of weak assumptions such that all the instance factors are not compromised

only if the intruder compromises all the instance factors he is able to impersonate the patient

<table>
<thead>
<tr>
<th>Removed Weak Asm(s)</th>
<th>Compromised Factors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PIN</td>
</tr>
<tr>
<td>BA1</td>
<td>x</td>
</tr>
<tr>
<td>BA2</td>
<td>✓</td>
</tr>
<tr>
<td>UBA1_Var1</td>
<td>✓</td>
</tr>
<tr>
<td>UBA2_Var1</td>
<td>x</td>
</tr>
</tbody>
</table>
Security Analyses

Analysis 1: Is the solution secure under all the strong and weak assumptions?

SATMC does not find any attack on the solution (i.e. the intruder is not able to impersonate the user)

Analysis 2: Which assumptions can be removed? (e.g., modeling a wrong implementation)

- none of the strong assumptions
- one weak assumption at a time
- a combination of weak assumptions such that all the instance factors are not compromised

only if the intruder compromises all the instance factors he is able to impersonate the patient
Phase 3: Security Analysis

1. Application Context
2. Customization of mID(OTP)
3. Security Analysis
4. Usability Analysis

Proximity Intruder

STOLEN SMARTPHONE

PIN
Phase 3: Output

MSC Scenario

Asms Scenario

Goals Scenario

Phase 4
Phase 4: Usability Analysis

- Monitoring apps require a daily or even hourly use
- Mobile keyboards are small and sometimes uncomfortable to use.

The designed solution:

- does not ask Patient to enter the OTP; after the PIN input, the OTP value is sent to ADC in a transparent way.

- provides a SSO experience. Until the session is valid, Patient has to digit only her PIN to access TreC or other federated apps.
TreC DEMO - Exploitation Phase
Outline

- Introduction and Problem Statement
  - Question 1: Mobile vs Browser-based Authentication
- Design Choices: Security and Usability Problems
  - Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen
- Methodology Overview: TreC Scenario
  - Question 2: e-health legal compliance
- Usability Discussion on TreC
  - Exercise 3: TreC activation phase
- Conclusions and On-going/Future Work
  - Question 3: Pros & Cons of our methodology and TreC solution
  - Question 1bis: Mobile vs Browser-based Authentication
Usability Problems

The activation phase is too complex:

1. It requires the use of a smartcard reader
   - need for an installed software
   - some browser incompatibility

2. The users are bothered by the use of a complex password
   - easily forgettable
How will you solve these two usability problems?
1.a Laptop  Using a portal made available by ADC, User logs in with a LoA 2 and obtains an QR code.

1.b Help Desk  A user, after an in-person identification, obtains an QR code.
Activation of OTP-PAT: 2nd Solution

2 Mobile

On her mobile, User scan the QR code using OTP-PAT, enters a temporary code obtained on her email and generates her PIN.
Introduction and Problem Statement

Question 1: Mobile vs Browser-based Authentication

Design Choices: Security and Usability Problems

Exercise 1: embedded browser  Exercise 2: OTP displayed on the screen

Methodology Overview: TreC Scenario

Question 2: e-health legal compliance

Usability Discussion on TreC

Exercise 3: TreC activation phase

Conclusions and On-going/Future Work

Question 3: Pros & Cons of our methodology and TreC solution

Question 1bis: Mobile vs Browser-based Authentication
Conclusions and On-going/Future Work

- New methodology for the design and security assessment of mobile authentication and SSO solutions

- Covered aspects:
  - Security
  - Usability
  - Legal-provisioning
  - SSO
  - MFA
  - Native apps

- Real-world scenarios: TreC ...

On-going - Future Work:

- Semi-automatic code generation
- Extensions of the AuthN aspects
- Formalization of other OTP generation approaches
Pros and Cons of our methodology and of the TreC solution?
Can we use web app authentication solutions for mobile apps?
Our Publications

• G. Sciarretta and A. Armando and R. Carbone and S. Ranise. 

• G. Sciarretta, R. Carbone and S. Ranise. 
  *A delegated authorization solution for smart-city mobile applications.*

• G. Sciarretta, R. Carbone, S. Ranise and A. Armando. 
  *Anatomy of the Facebook solution for mobile single sign-on: Security assessment and improvements.*

• G. Sciarretta, R. Carbone, S. Ranise and L. Viganò. 

https://st.fbk.eu/publications/POST-2018
Thanks for your attention!

Acknowledgements. This work has partially been supported by the activity “STAnD” of the action line Digital Infrastructure of the EIT Digital.

Any questions?

https://st.fbk.eu/workshop-ifipsc-18

ranise@fbk.eu - giada.sciarretta@fbk.eu