The Cluster Exposure Verification (CLÉA) Protocol

https://github.com/TousAntiCovid/CLEA-exposure-verification

V. Roca, A. Boutet, C. Castelluccia (Inria, PRIVATICS team, FR)

March 23rd, 2021

Goals

- focus on **public/commercial locations** (restaurant, bar, sport center, show, train, shared ride) and **private events** (wedding, private party)

- **easy “check-in”** to a location, by scanning a QR code,

  or filling a hand-written register, according to user’s preference

example 65x65,
level 12, QR code

- automatically **detect** potential cluster locations/events

- automatically **notify** a user who shared, at the same time, a location/event with one or more COVID+ users

Two key design choices, for good reasons (privacy, automatic functioning)

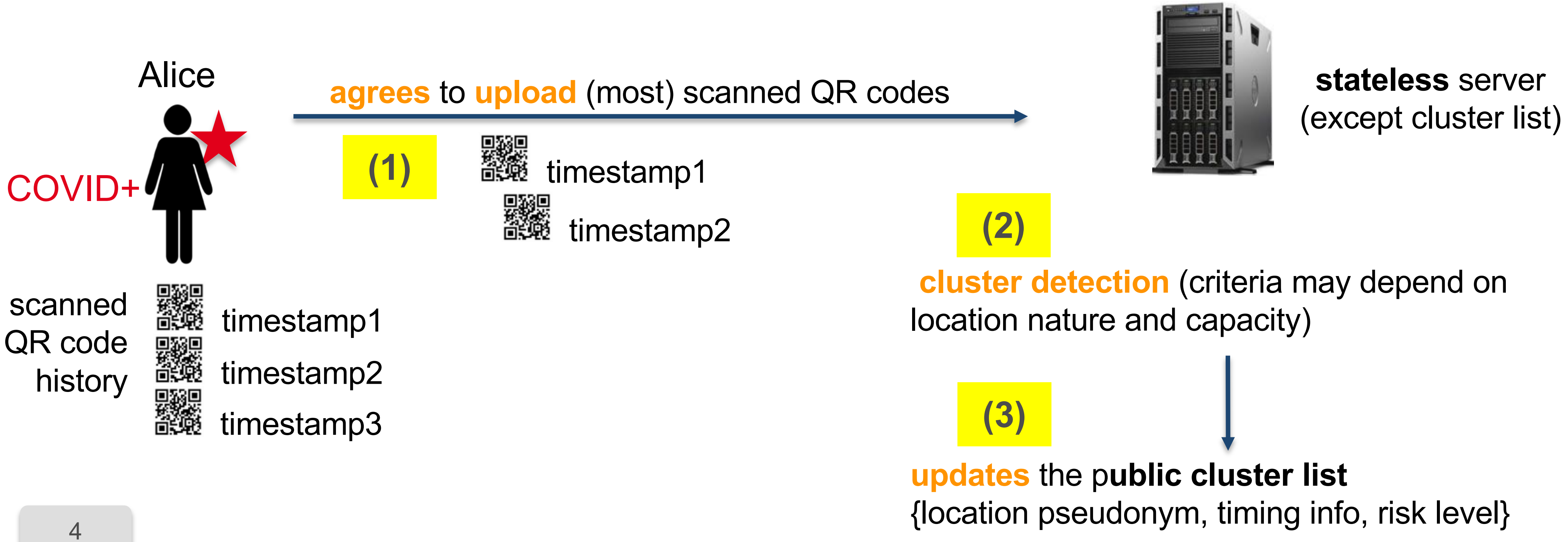
#1: centralized cluster detection

#2: decentralized risk estimation and notification

and a direct consequence, a **public list** of cluster pseudonyms

Key choice #1: centralized cluster detection

- COVID+ users are invited to upload their scanned QR codes + timing
- server detects clusters and updates {location pseudonym, timestamp + duration} cluster list



Alice

COVID+

scanned QR code history

timestamp1

timestamp2

timestamp3

agrees to **upload** (most) scanned QR codes

(1)

timestamp1

timestamp2

stateless server (except cluster list)

(2)

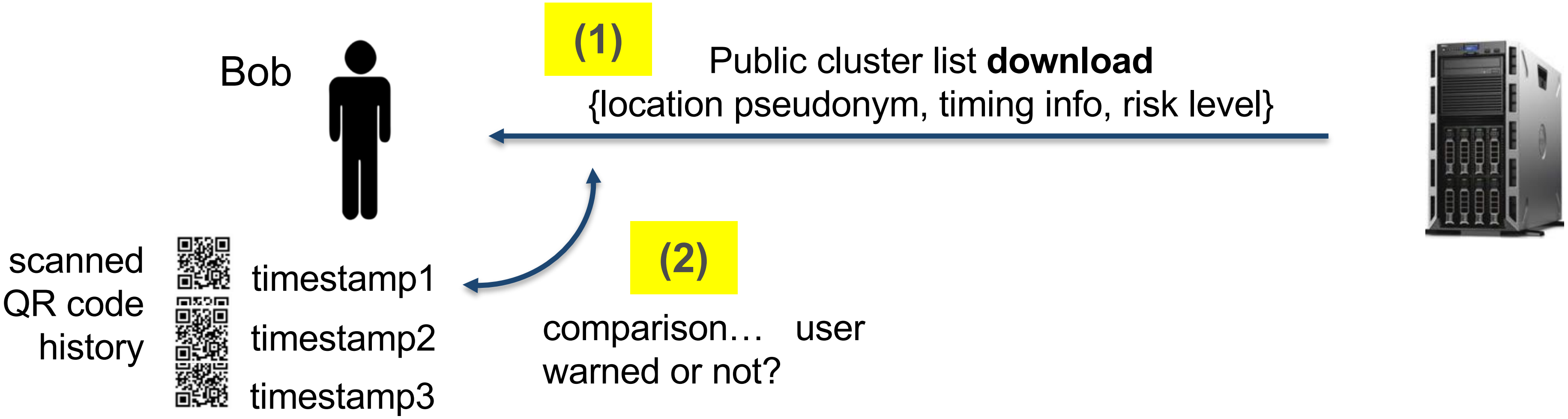
cluster detection (criteria may depend on location nature and capacity)

(3)

updates the **public cluster list** {location pseudonym, timing info, risk level}

Key choice #2: decentralized risk estimation/notif.

- scanned QR codes remain on the user smartphone (if not tested COVID+)
- compares scanned QR codes with the cluster list info



Bob

(1) Public cluster list **download**
{location pseudonym, timing info, risk level}

(2)

comparison… user warned or not?

scanned QR code history

timestamp1

timestamp2

timestamp3

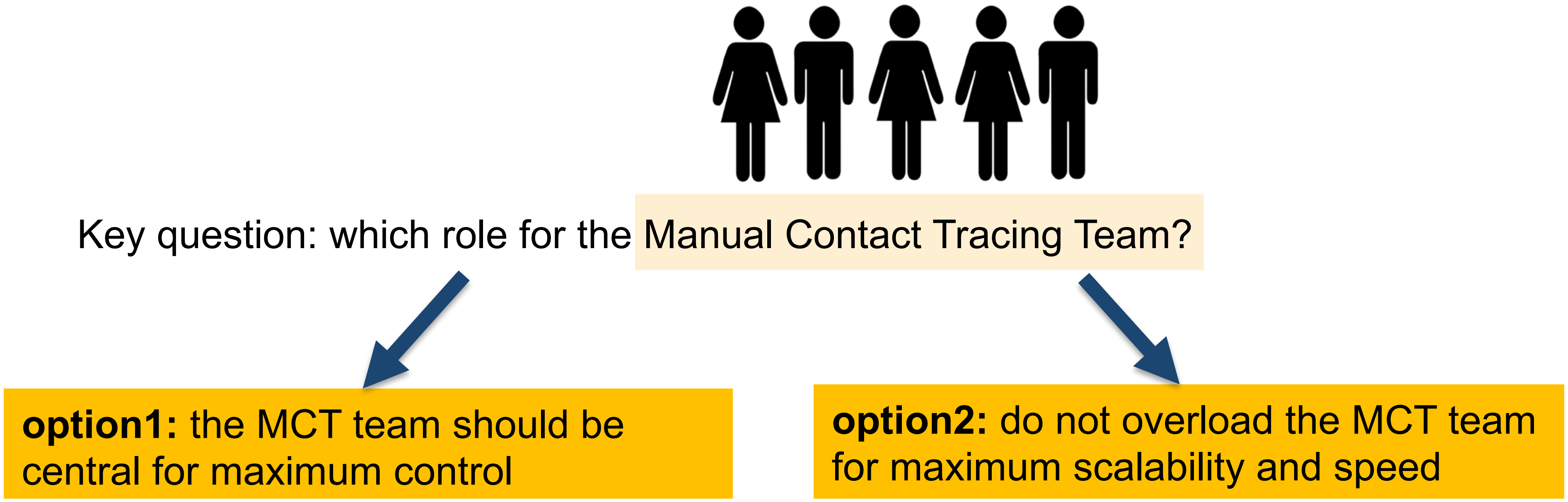
Key choice #2: decentralized risk estimation/notif. (2)

- decentralized risk analysis requires sharing cluster list: {loc. pseudos; timing info; risk level}
  - this is **not** sensitive medical data per se
  - with dynamic QR codes, pseudonyms are temporary ☺

situation totally different from **contact tracing** where decentralized risk analysis (e.g., GAEN) requires to share publicly the pseudonyms of users tested COVID+

- it's **sensitive health** data ☹, and anyone can easily know if a neighbor is COVID+
  - see: https://coronadetective.eu
- GAEN is not very GDPR friendly

A **single** protocol, CLÉA, **two** potential deployments

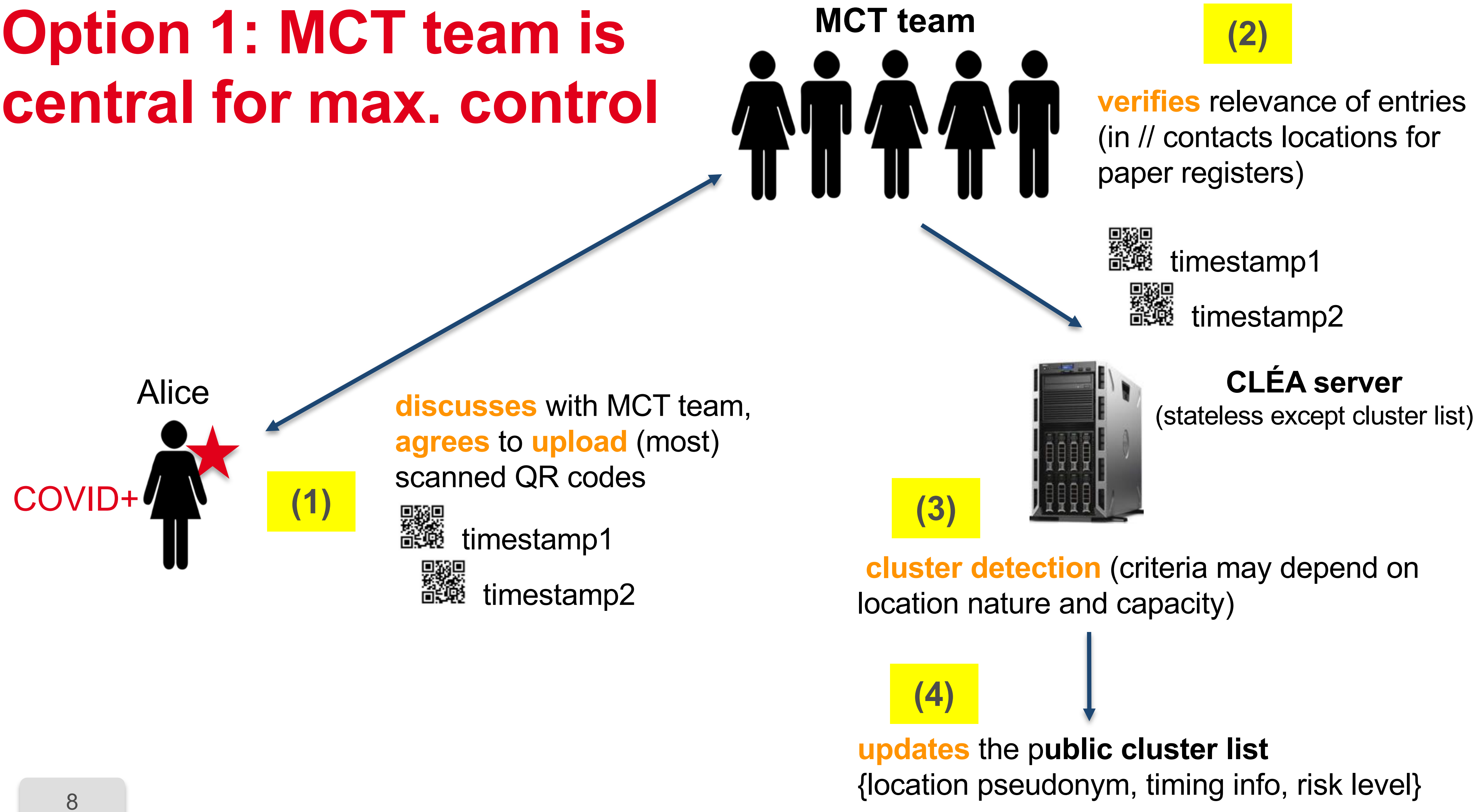


Key question: which role for the Manual Contact Tracing Team?

option1: the MCT team should be central for maximum control

option2: do not overload the MCT team for maximum scalability and speed

Option 1: MCT team is central for max. control



MCT team

(2)

verifies relevance of entries (in // contacts locations for paper registers)

timestamp1

timestamp2

CLÉA server

(stateless except cluster list)

Alice

COVID+

(1)

discusses with MCT team, **agrees** to **upload** (most) scanned QR codes

timestamp1

timestamp2

(3)

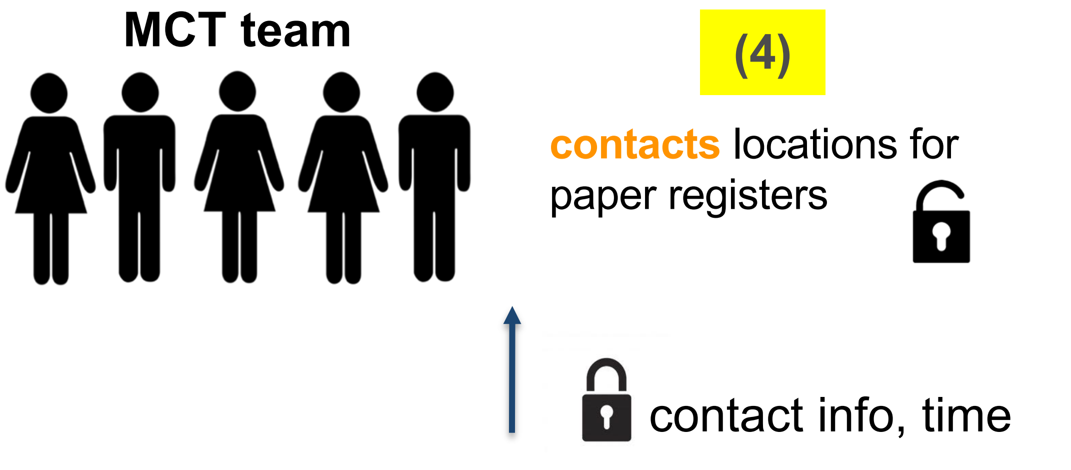
cluster detection (criteria may depend on location nature and capacity)

(4)

updates the p**ublic cluster list**

{location pseudonym, timing info, risk level}

Option 2: do not overload the MCT team for max. scalability and speed

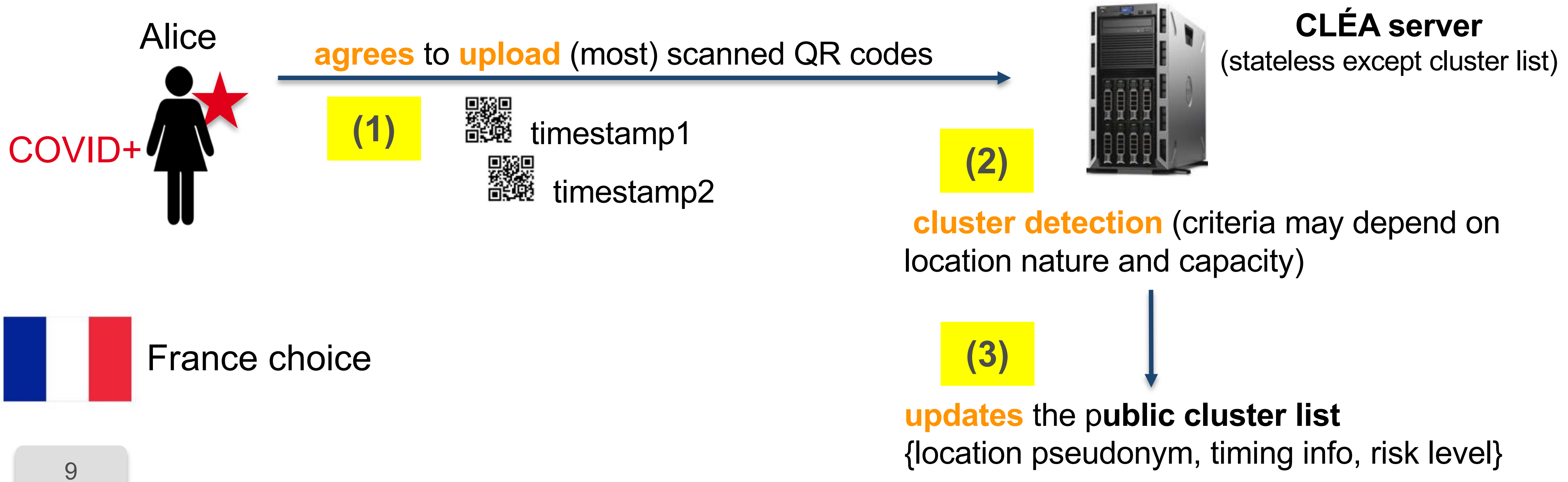


MCT team

(4)

contacts locations for paper registers

contact info, time



Alice

COVID+

agrees to upload (most) scanned QR codes

(1)

timestamp1

timestamp2

CLÉA server

(stateless except cluster list)

(2)

cluster detection (criteria may depend on location nature and capacity)

(3)

updates the public cluster list

{location pseudonym, timing info, risk level}

France choice

Additional technical considerations:

- structure of a QR code
- static or dynamic QR code?
- compatible with dedicated ~~plug~~-and-play devices

Structure of a QR code

- a 65x65 (level 12, M or Q redundancy) QR code

- contains a **“deep-link” URL**
  - example (FR): https://tac.gouv.fr/O9QAalpq3qpQP…N2qpcAA0dmaCQ

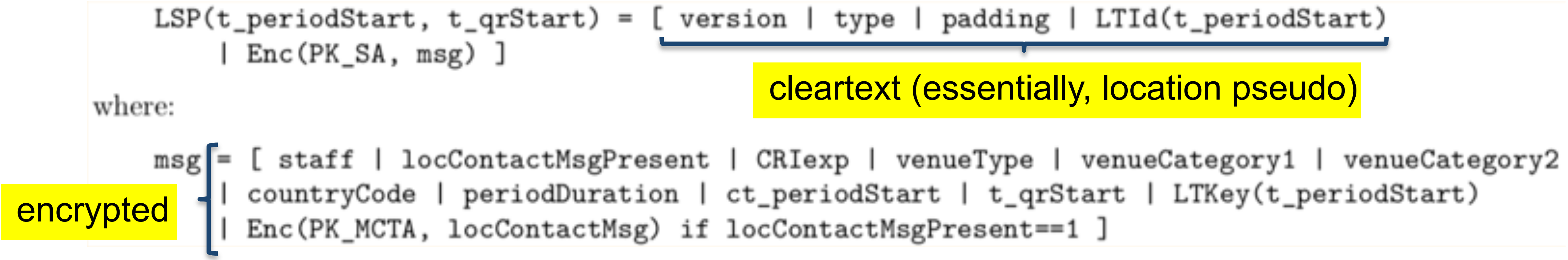
country specific prefix

location specific dynamic suffix

- a scanned QR code
  - is either automatically managed by the CLÉA application (if installed)
  - otherwise user is redirected to the https://tac.gouv.fr web site

Structure of a QR code (2)

- location specific suffix
  - **cleartext part:** essentially the location pseudonym (Location Temporary ID)
  - **encrypted part:** essentially the location key, plus location typology, and encrypted location contact information



```
    LSP(t_periodStart, t_qrStart) = [ version | type | padding | LTId(t_periodStart)
        | Enc(PK_SA, msg) ]
where:

    msg = [ staff | locContactMsgPresent | CRIexp | venueType | venueCategory1 | venueCategory2
        | countryCode | periodDuration | ct_periodStart | t_qrStart | LTKey(t_periodStart)
        | Enc(PK_MCTA, locContactMsg) if locContactMsgPresent==1 ]
```

cleartext (essentially, location pseudo)

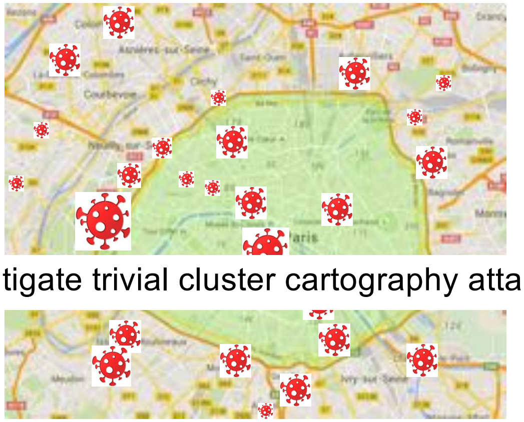
encrypted

- LSP is Base64 encoded and included in the deep link: https://tac.gouv.fr/<base64(LSP)>

Static or dynamic QR code?

- **issue:** we have a **public** cluster list {cluster location pseudonyms + timing info}

- **dynamic QR codes:**
  - mitigate trivial cluster cartography attacks, since pseudonyms change all the time
  - makes replay attacks a bit more complex, since QR codes have a limited time validity
  - improves user privacy, since 2 COVID+ users at the same location across two different days cannot be linked (upload different location pseudonyms)

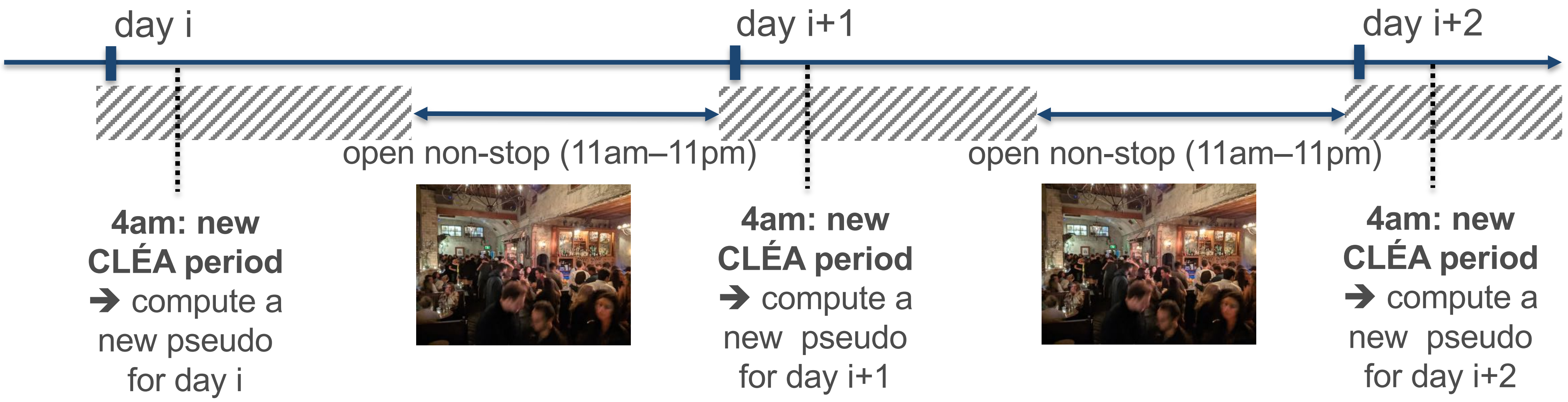


Mitigate trivial cluster cartography attack

- try to be **as secure as possible**, although no full-proof guaranties

Static or dynamic QR code? (2)

- example: compute a location temporary pseudonym (LTId) per day



day i

day i+1

day i+2

open non-stop (11am–11pm)

open non-stop (11am–11pm)

4am: new CLÉA period
➔ compute a new pseudo for day i

4am: new CLÉA period
➔ compute a new pseudo for day i+1

4am: new CLÉA period
➔ compute a new pseudo for day i+2

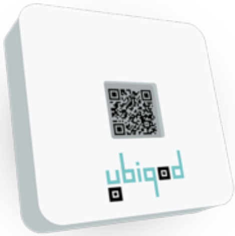
```
LTKey(t_periodStart) = SHA256(SK_L | t_periodStart)
LTId(t_periodStart) = HMAC-SHA-256-128(LTKey(t_periodStart), "1")
```

Compatible with dedicated ~~plug~~-and-play devices

- an **easy to deploy** solution for public/commercial locations and **dynamic** QR codes
  - an option, not an obligation
- pre-configured, install-and-forget **commercial devices** (e-ink 200*200pix. display)
  - no onsite configuration, comes ready to use
  - no wireless connection / power plug / USB connector / button
- **one or more devices** per location, depending on size
  - all devices compute the same location pseudonym

static, printed QR code

OR



dynamic QR code

https://www.skiply.eu/ubiqod-key/

Important particular cases:

1. employees
2. private events
3. linking CLÉA and hand-written attendance register
4. pan-European interoperability

Particular case #1: the location employees

- employees must benefit from CLÉA (be warned if the work place is cluster)
  - major difference: a employee stays in the location for longer periods than a client
  - since scanning every 2 hours is not a solution, a device can produce a “Staff” QR code
    - a “Staff” QR code is valid till the end of current period
    - (NB: a magnetic detector on a Skiply device enables to produce a “Staff” QR code)
  - the employee CLÉA app recognizes the “Staff” QR code and its extended duration to assess risks
- an employee tested COVID+ should be able to upload her scanned QR code history
  - the CLÉA server recognizes the “Staff” QR code and its extended duration to assess risks
- easy to address ☺

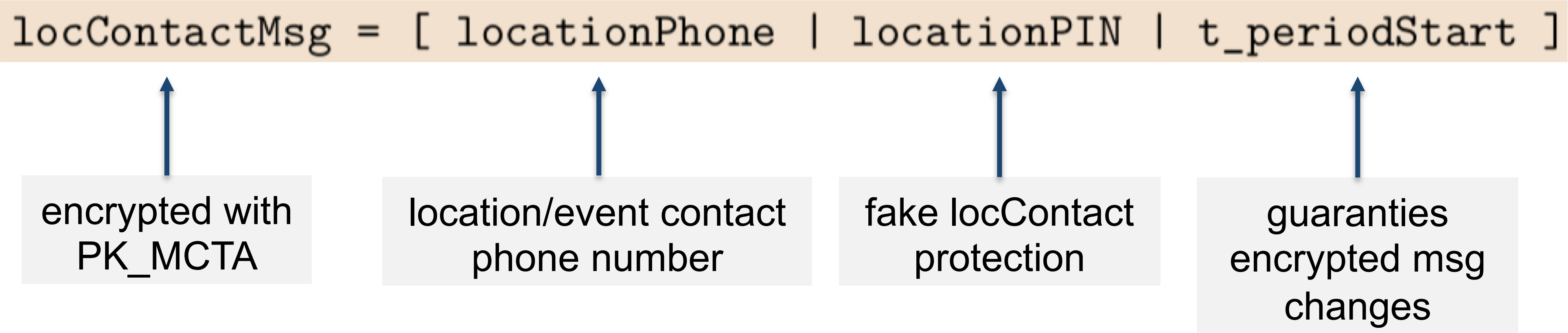
Particular case #2: private events

- choose a **static, printed QR code**
  - to be generated on a Web service, printed and displayed at the entrance
  - it’s necessarily a static, time limited (for this event) QR code
  - event may last more than a single day…
- NB: a location that does not care about cluster cartography attacks may also opt for a static, printed QR code

The same CLÉA system handles both static and dynamic QR codes the same way (no protocol change)

Case #3: Linking CLÉA and hand-written registry

- CASE 1: a user tested COVID+ has used the CLÉA system
  - a link is necessary to inform the location/event manager, get the registry, inform others
  - QR code contains encrypted contact information



```
locContactMsg = [ locationPhone | locationPIN | t_periodStart ]
```

encrypted with PK_MCTA

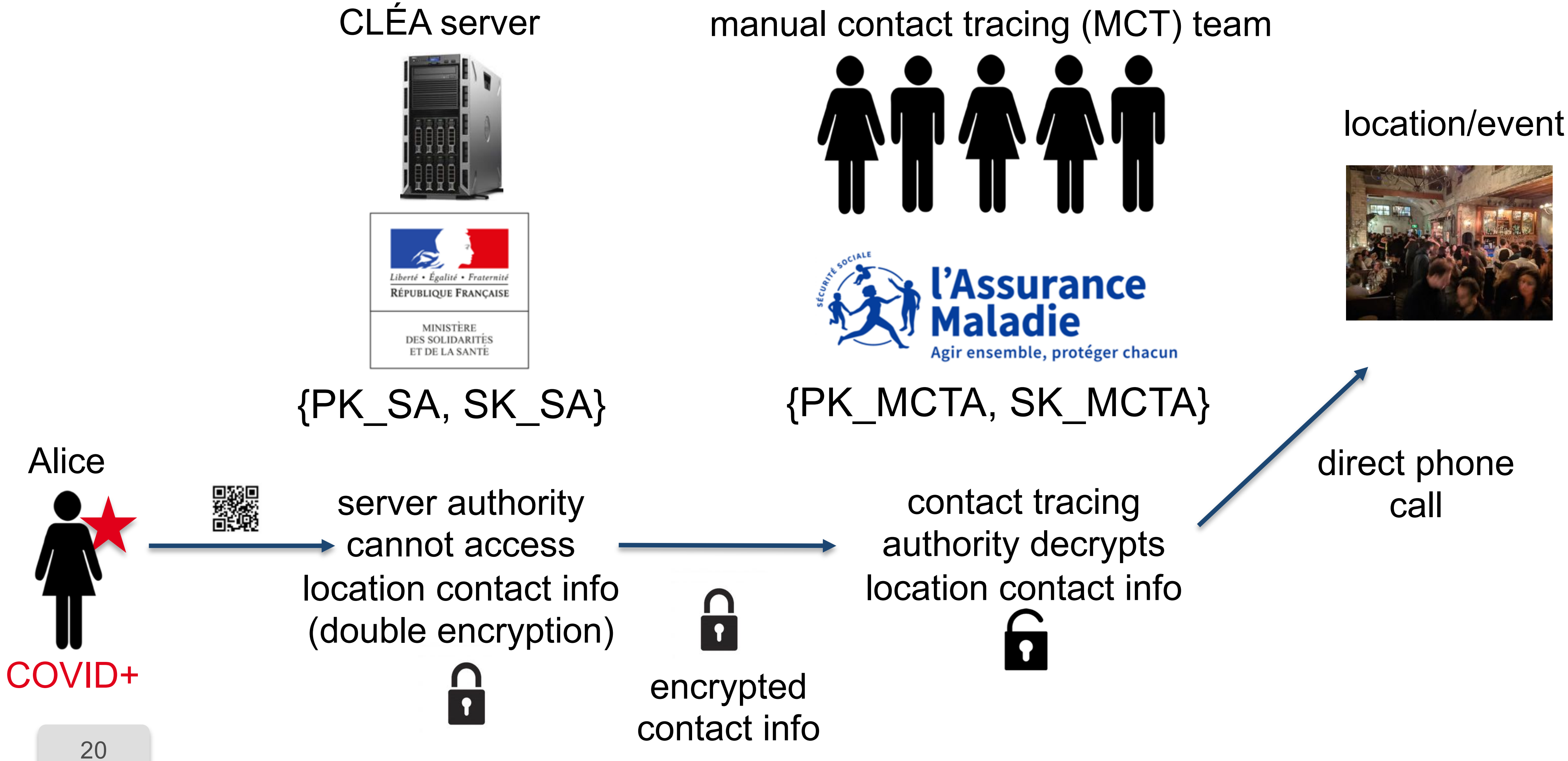
location/event contact phone number

fake locContact protection

guaranties encrypted msg changes

  - only the Contact Tracing Team Authority can decrypt it (it's a different authority)

Linking CLÉA and hand-written registry (2)



CLÉA server

Liberté • Égalité • Fraternité
République Française
Ministère des Solidarités et de la Santé

{PK_SA, SK_SA}

manual contact tracing (MCT) team

Sécurité sociale
l'Assurance Maladie
Agir ensemble, protéger chacun

{PK_MCTA, SK_MCTA}

location/event

Alice

COVID+

server authority cannot access location contact info (double encryption)

encrypted contact info

contact tracing authority decrypts location contact info

direct phone call

Linking CLÉA and hand-written registry (3)

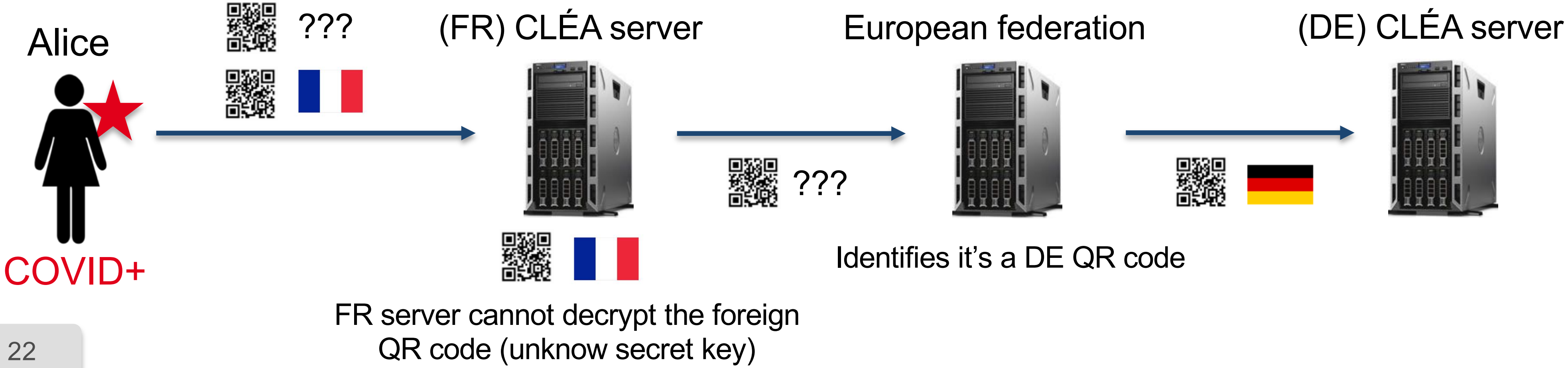
- CASE 2: a user tested COVID+ has used the hand-written registry
  - *assumption1: the user remembers having been to a location and when*
  - the MCT team asks the location/event contact to send the paper attendance registry…
  - … and a QR code generated that day
  - *assumption 2: the location contact has scanned a QR code that day*
  - the location contact uploads the QR code, the MCT team identifies the location pseudonym used that day, and can inform the CLÉA server (details TBD)

- involves a few risks, yet seems realistic
  - because the location contact person has a personal interest in scanning QR codes

Particular case #4: pan-European interoperability

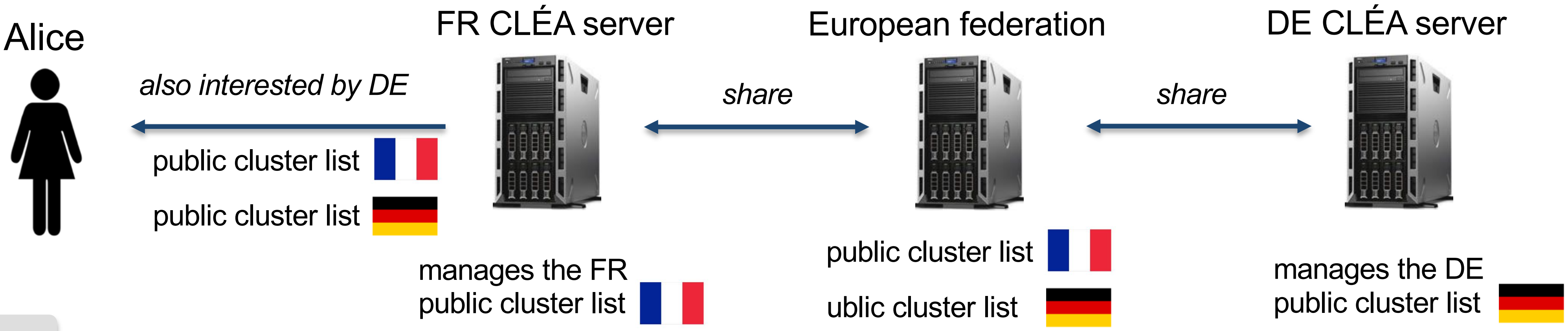
- QR codes do include 3-digit ISO 3166-1 country code
  - e.g., 250 for France
  - in the encrypted part of the QR code

- Example: Alice, who went to a German restaurant, is tested COVID+ and agrees to share her scanned QR codes…


Alice
COVID+
???
(FR) CLÉA server
European federation
(DE) CLÉA server
???
Identifies it's a DE QR code
FR server cannot decrypt the foreign QR code (unknow secret key)

Particular case #4: pan-European interoperability (2)

- National servers also need to share their public cluster list
- Example: Alice, who went to a German restaurant, wants to know if she is at risk…



Summary

CLÉA benefits

- **Speed-up** notifications because **“time is key”**
  - each scanned QR code is immediately usable by the CLÉA server

  (because scanned QR codes are self-sufficient ☺)

- **Minimize MCT team work** in the critical path (even with option 1)
  - cluster qualification/user notification is automatic
  - no need to search a phone number and contact the location/event manager

  (because scanned QR codes are self-sufficient ☺)

CLÉA benefits (2)

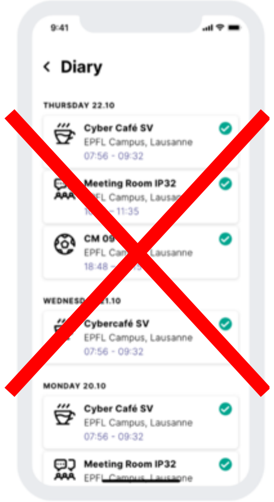
- Minimize practical risks for **maximum reliability**
  - users no risk to “forget” inadvertently visited locations

  (because scanned QR codes are self-sufficient ☺)

?

- Preserve **user privacy** as much as possible
  - manipulate, store, send location (rotating) **pseudonyms only**
  - never store real location names and addresses!

  (because scanned QR codes are self-sufficient ☺)



9:41

Diary

THURSDAY 22.10

Cyber Café SV
EPFL Campus, Lausanne
07:56 - 09:32

Meeting Room IP32
FL Campus, Lausan
- 11:35

CM 09
EPFL Can Lausanne
18:48 -

WEDNESD 1.10

Cybercafé SV
EPFL Campus, Lausanne
07:56 - 09:32

MONDAY 20.10

Cyber Café SV
EPFL Campus, Lausanne
07:56 - 09:32

Meeting Room IP32
EPFL Campus, Lausanne

CLÉA benefits (3)

- **Reduce risks of attacks** by asking a “proof of presence” in a location
    - although not perfect, a location cannot be qualified cluster unless a valid scanned QR code is exhibited

example 65x65, level 12, QR code

- Enable efficient **interoperability** across borders
    - a Country Code for efficient routing of QR codes
    - accommodates different national deployment choices

- fast, practical, flexible, interoperable, natively designed for presence tracing and cluster detection

- to be added **mid-April** to our French TousAntiCovid app
  - *NB: added does not mean it’s used (depends on re-opening)*
  - CNIL and ANSSI reviews under progress

- documents and open-source code
  - https://gitlab.inria.fr/stopcovid19/CLEA-exposure-verification
  - https://github.com/TousAntiCovid/CLEA-exposure-verification (github mirror)

Thank you…

Contact: vincent.roca@inria.fr