

When to change **the locks**

THE TEAM

Hello **again.**

01



David Lassig
Platform Product Owner Backend · BWI

02



Timm Behner
Security Engineer (Cryptography) · BWI

03



Björn Reiners
Product Manager & Storyteller · BWI

OUR PRODUCTS

SEGIMER

ARMINIUS

Where do we stand **today with MLS?**

MSC4256

- RFC 9420 MLS mode Matrix
- Membership shifted into client-side MLS group state
- Opened · 24 Jan 2025 · **636 days** ago
- Reviewed · 88 comments · 45 reviews

Now

- Continues foundational ideas of MSC4256
- Removes some tradeoffs of MSC4256 in favor of MLS
- MLS wraps around Matrix with additional services and keeps the Matrix spec unchanged
- Unlinkability still necessary and needs MSCs

WHAT BROKE

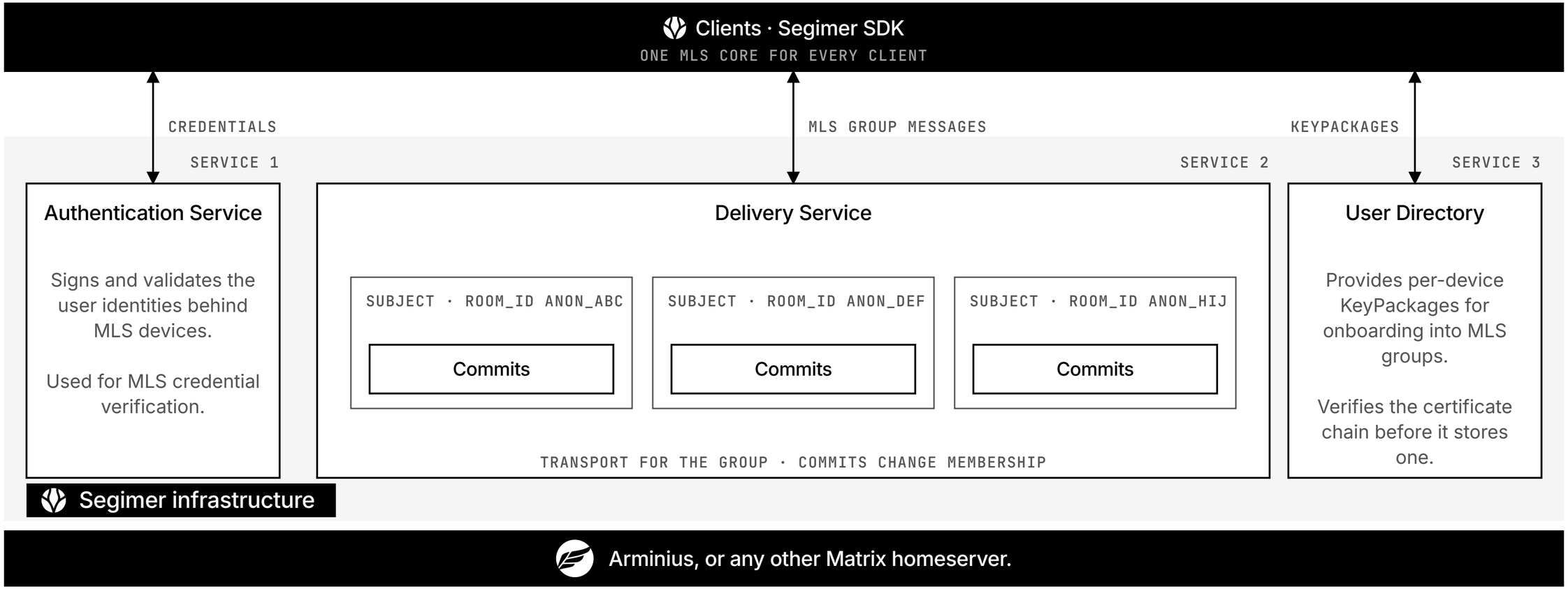
Nothing worked the **first time.**

Fitting the MLS lifecycle into the Matrix lifecycle looked impossible.

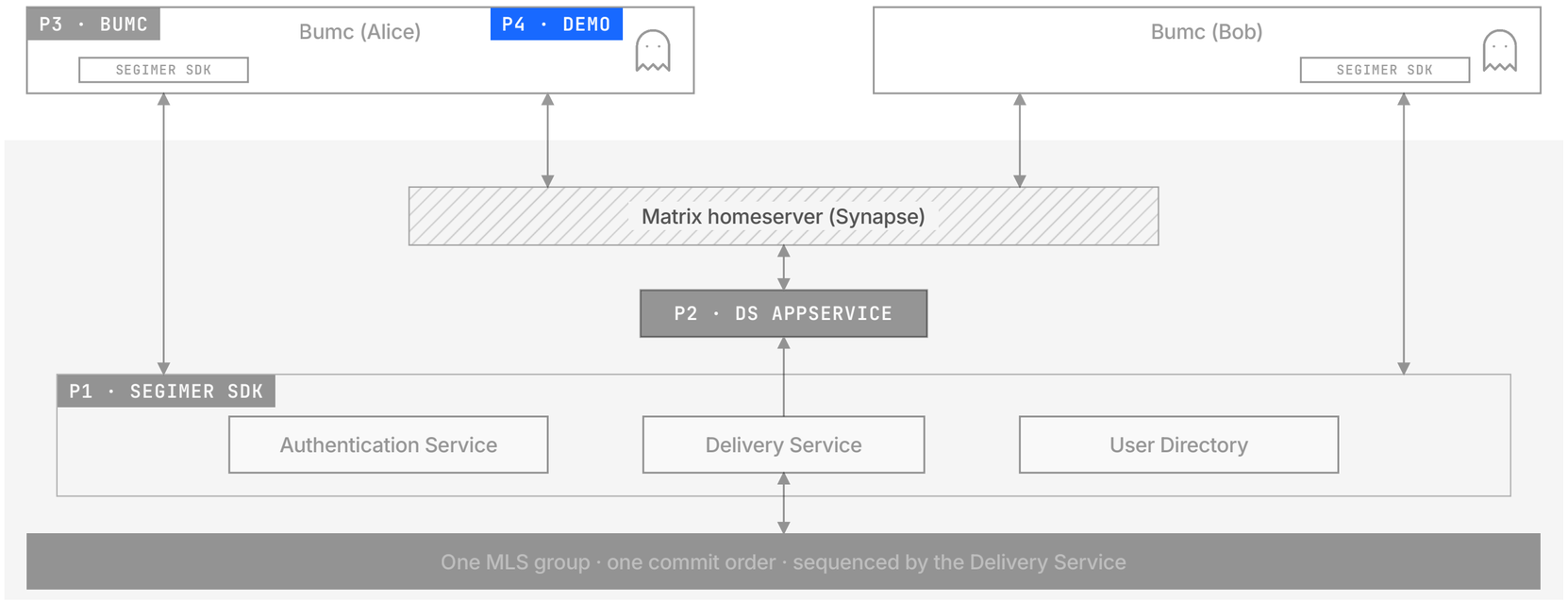
MLS		Matrix
New member **per group**	DEVICES	Same device **in every room**
ID from **the client**	IDENTITY	One global ID **from the server**

Unlinkability is **the goal.**

So we got to work and built **Segimer.**



Our showcase for today, **built on top of Segimer.**



P1 · SEGIMER SDK

MLS for every client

P2 · DS APPSERVICE

MLS groups, mirrored into Matrix

P3 · BUMC

The new desktop client

P4 · DEMO

Live, on two clients

P1

PIECE ONE · SEGIMER SDK

MLS, **made easy.**

The Segimer SDK in your client: two hooks, and it speaks MLS.

Two hooks. **That is the plugin.**

The Matrix Rust SDK gets two hooks: encrypt before send, decrypt after sync. The Segimer SDK implements both. Everything else stays Matrix.

01 Keep changes in the Matrix Rust SDK to a minimum

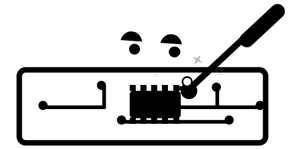
02 Build the Matrix Rust SDK without Segimer

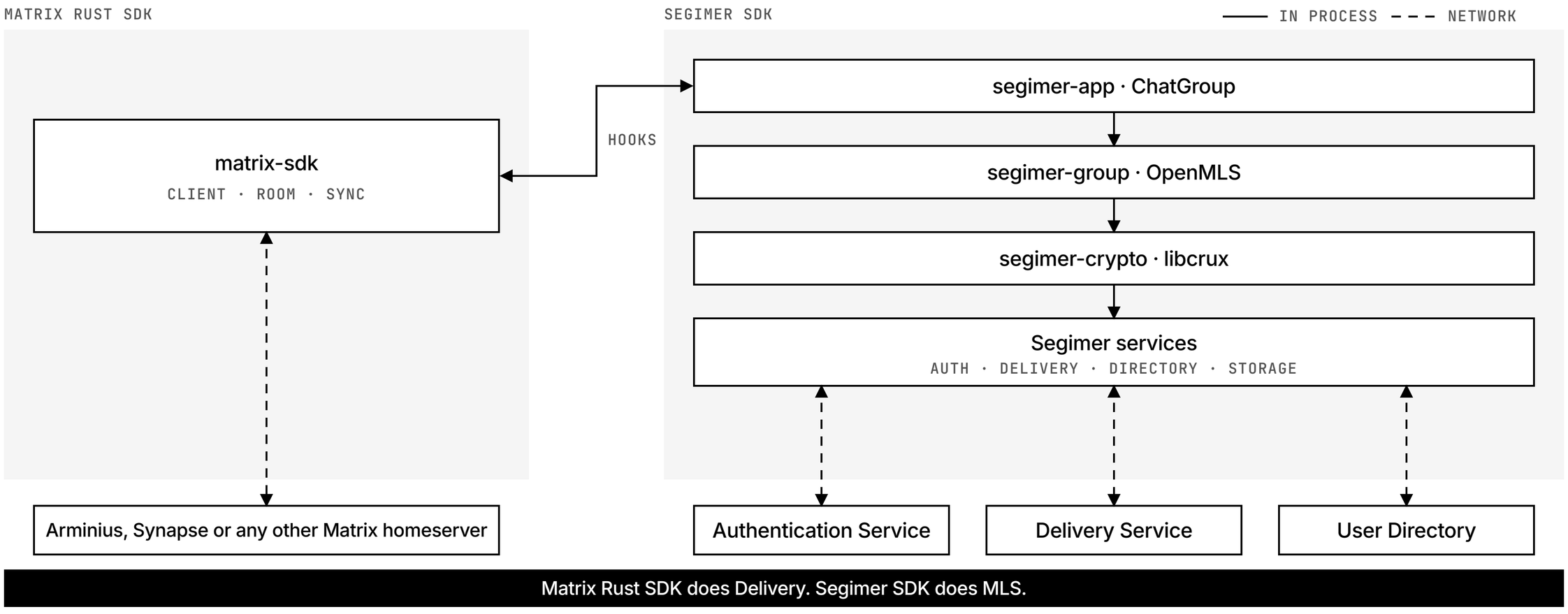
03 Test both sides in isolation

```rust
// Matrix Rust SDK · extension API
trait EncryptHook {
    fn encrypt(
        &self,
        room: &RoomId,
        event: &Raw<AnyMessageLikeEventContent>,
    ) -> Result<Option<Encrypted>>;
}

trait DecryptHook {
    fn decrypt(
        &self,
        room: &RoomId,
        encrypted: &EncryptedRoomMessageEventContent,
    ) -> Result<Decrypted, UnableToDecryptReason>;
}
```

Use MLS **like a plugin.**





Matrix Rust SDK does Delivery. Segimer SDK does MLS.

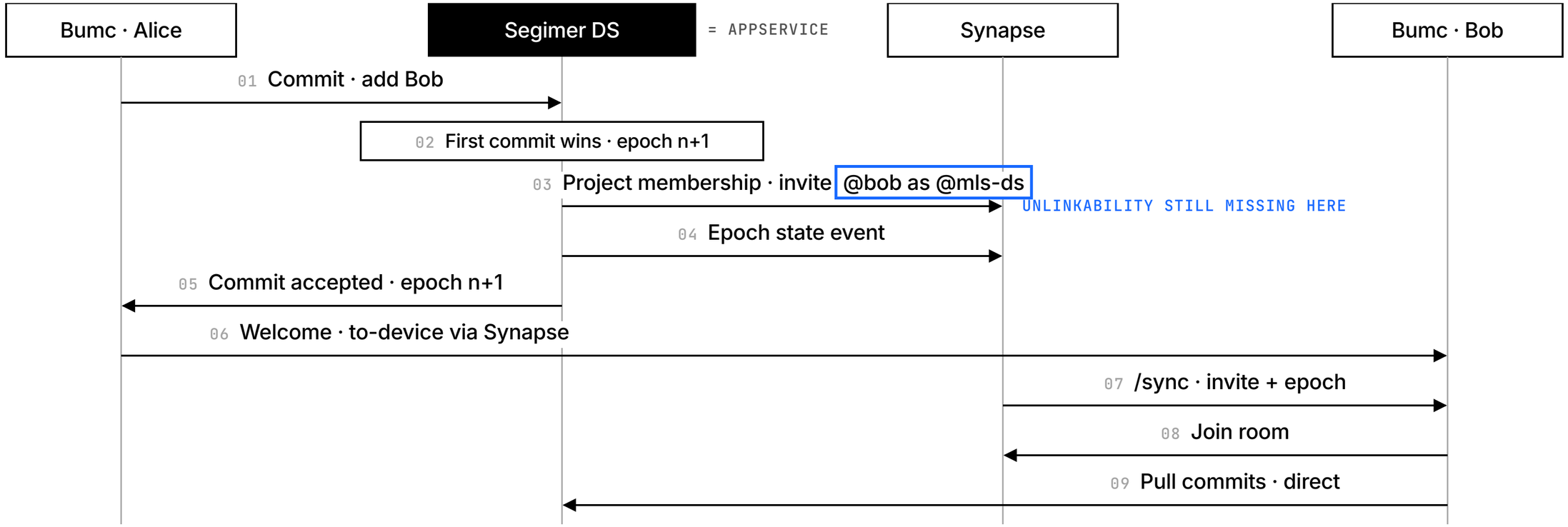
P2

PIECE TWO · DS APPSERVICE

DS AppService.

The Delivery Service, seen by the homeserver as an Application Service.

Membership **with MLS.**

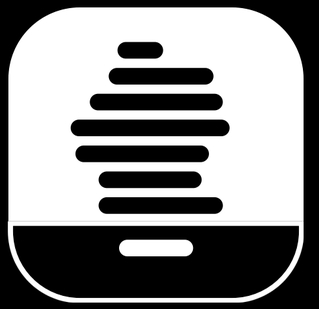


Membership is projected one way: the MLS group is the source, the Matrix room a superset.

P3

PIECE THREE · BUMC

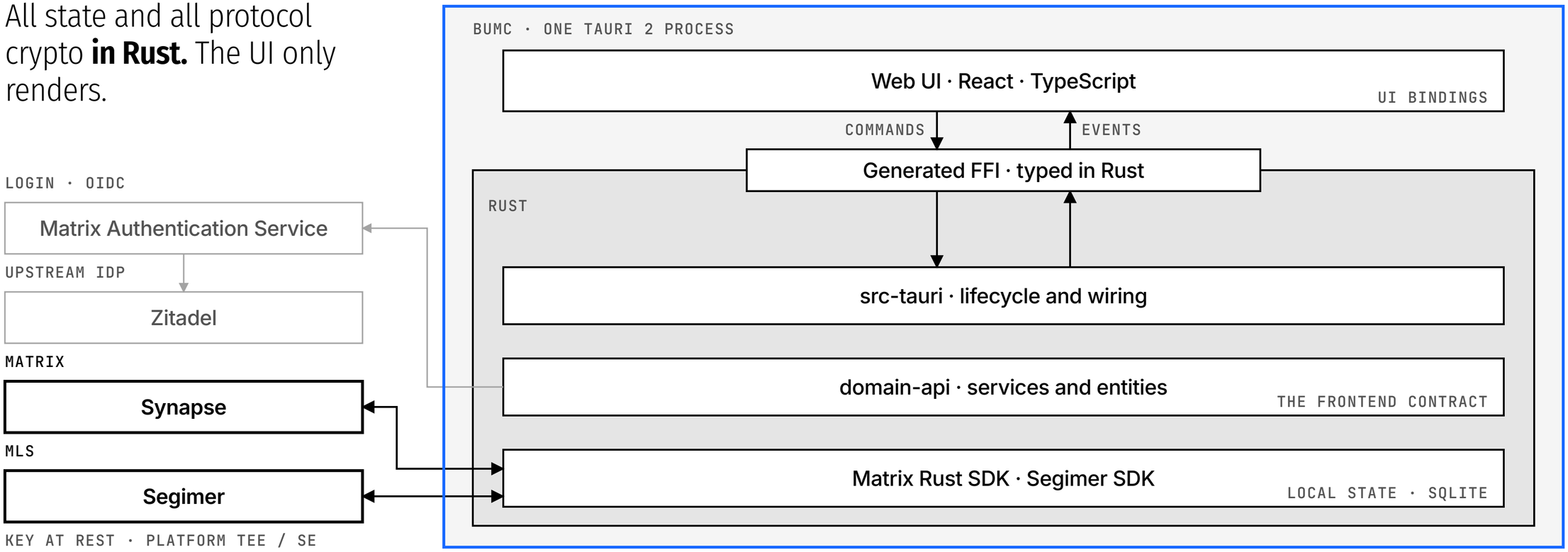
A new client, a thin UI.



Built from scratch, with room to grow.

Rust underneath. **We call it Bumc**

All state and all protocol crypto **in Rust.** The UI only renders.

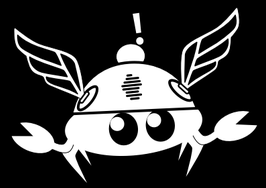


P4

PIECE FOUR · DEMO TIME

Let’s demo: **Incognito** mode.

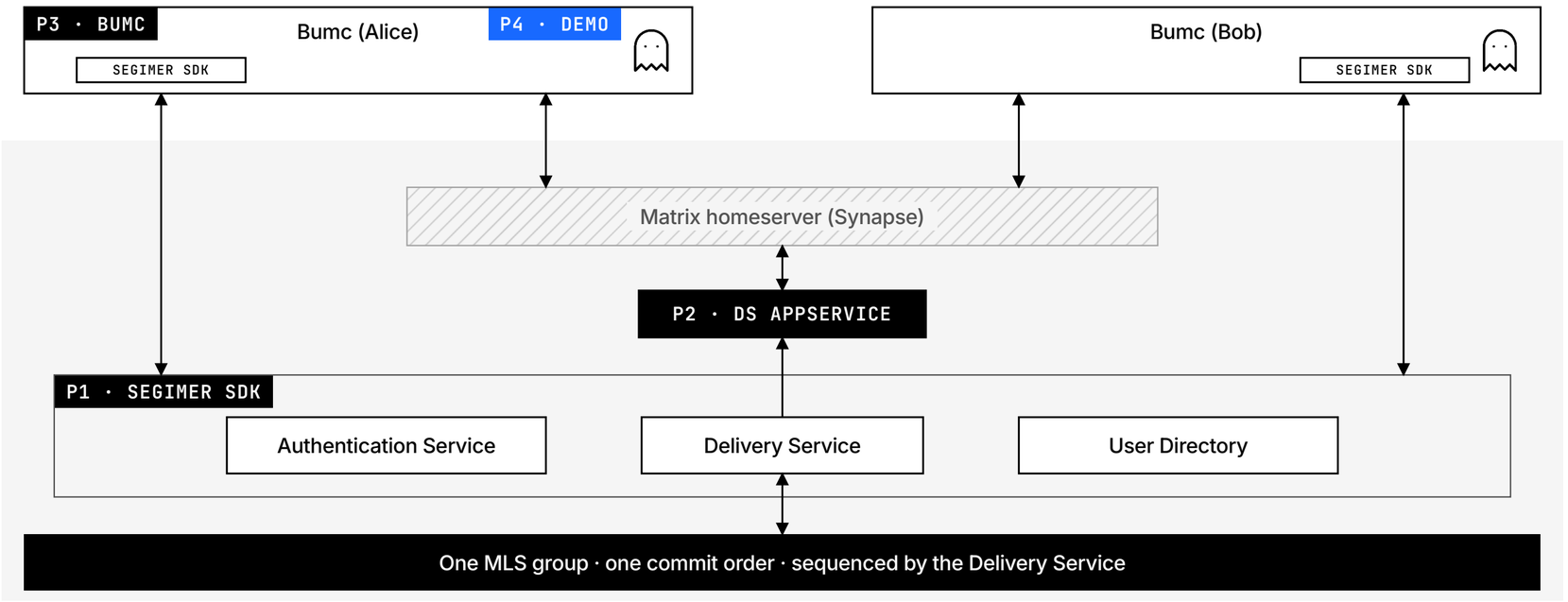
- Incognito mode: ephemeral rooms, destroyed on schedule
- MLS groups are destroyed client-side
- Matrix groups are destroyed via the DS application service bot
- Leave no trace: unlinkability still missing
- Once the MLS epoch closes, all the keys are also gone.



Nobody saw **a thing.**



What we are going **to show.**



P1 · SEGIMER SDK

MLS for every client

P2 · DS APPSERVICE

MLS groups, mirrored into Matrix

P3 · BUMC

The new desktop client

P4 · DEMO

Live, on two clients

THE ANSWER

When do you change the locks?
Every epoch.

STICKERS · FREE

Ask us anything.
Leave with a crab.

One question, one sticker. Find us after the talk.



THANKS.

Questions? **Please.**

FIND US ON SOCIALS

BWI
IT für Deutschland

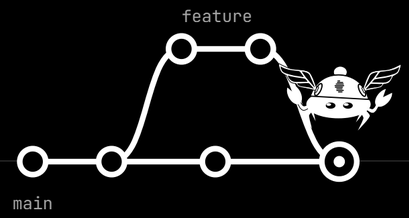
BundesMessenger is a product of BWI GmbH

LINKEDIN

THE CODE · APACHE 2.0

Arminius · Rust homeserver	AVAILABLE
DS adapter · Matrix and MLS	JAN 2027
Segimer · SDK and services	JAN 2027
Bumc · desktop client	JAN 2027

gitlab.com/bundesmessenger



ARCHITECTURE
David Lassig
Platform Product Owner Backend · BWI
david.lassig@bwi.de

CRYPTOGRAPHY
Timm Behner
Security Engineer (Cryptography) · BWI
timm.behner@bwi.de

PRODUCT
Björn Reiners
Product Manager & Storyteller · BWI
bjoern.reiners@bwi.de