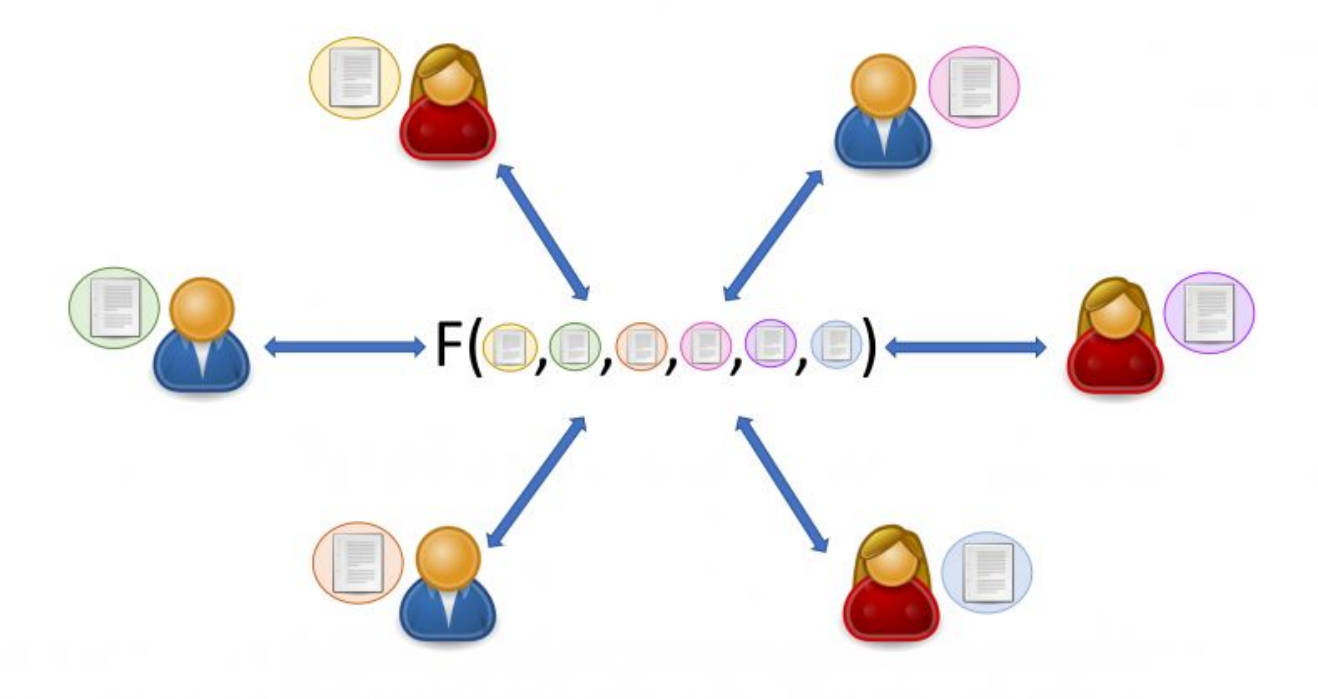
CURSIVE Vivek Bhupatiraju

Digital pheromones

MPC for human connection & coordination

What are digital pheromones?

What is multi-party computation?



What are biological pheromones?

A chemical substance produced by an animal and serves as a stimulus to other individuals of the same species for one or more behavioral responses

A fanning honeybee exposes Nasonov's gland (white – at tip of abdomen) releasing pheromone to entice swarm into an empty hive

Dogs communicate using pheromones and olfactory signals in urine.[19]

Male *Danaus chrysippus* showing the pheromone pouch and brush-like organ in Kerala, India

Digital pheromone principles

1) **Lightweight, privacy-preserving signals** we create for others to discover connection & perform coordination

2) **Fully programmable & verifiable** — you can choose what conditions you want to match on & require that important data is ZK proven

3) **Neutral peer-to-peer cryptographic protocol** — not tied to a specific app, doesn't require a server for coordination

Improves *discoverability*

Present

- At the whim of **AI algorithms and advertising markets**, who have all of your personal data
- **Unaligned objectives**: maximizing attention capture + money you spend

Improves *discoverability*

Present

- At the whim of **AI algorithms and advertising markets**, who have all of your personal data
- **Unaligned objectives**: maximizing attention capture + money you spend

Future

- Full ownership over our data with **signatures & ZK**
- Controllable + safe interfaces to learn more about your social graph with **MPC**

Improves *depth of connection*

Present

- No **verifiability on data**, more bots than humans
- All profiles are **public, sanitized billboards** of our lives & personality

Improves *depth of connection*

Present

- No **verifiability on data**, more bots than humans
- All profiles are **public, sanitized billboards** of our lives & personality

Future

- All info is **signed & proven**
- **Private data custody**, can share deeper personal info
- Can safely discover common & complementary traits through **PSI & MPC**

What could digital pheromones potentially enable?

Narrowcasting

The opposite of broadcasting.

We don’t need to be dependent on public feeds or group chats. ***Just narrowcast info to your most relevant connections!***

Unbreakable Consent

One of MPC's main flaws is that parties can exit protocol early.

In 2PC query-responses, this is a feature — ***full consent is mathematically required*** for querier to learn anything!

Superconnectors

Receive summaries of your friend's data that is privacy-preserving but you can compute with.

Instead of an algorithm, you can recommend two people or a group of people to meet based on synergy detection!

Cutting out the middleman

Don’t need social media feeds to collect personal data and sell it to businesses.

Can import & self-attest to data and get directly matched with businesses on your own terms

I’m Feeling Serendipitous

Walk into a public space and put out some people you would like to meet or activities you would want to do, get matched with others who want the same.

Privately manifest people and things you’d like to do in the moment!

Love Thy Neighbor

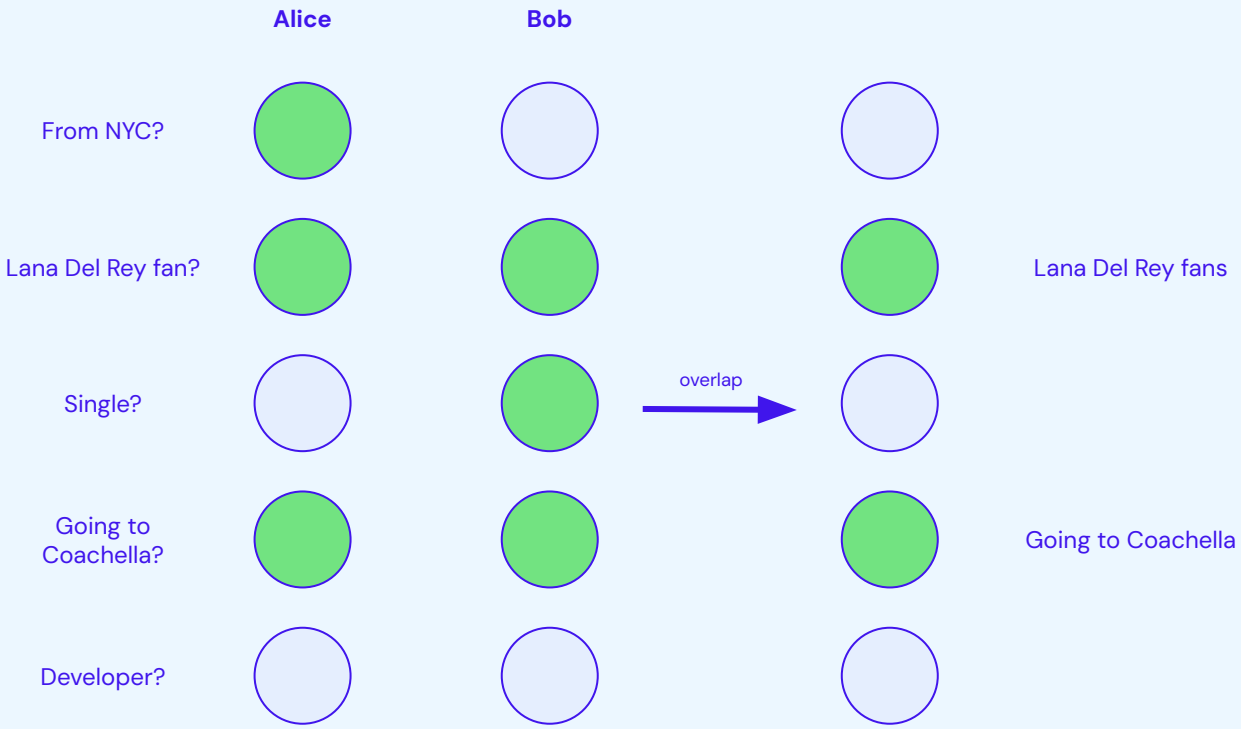
Discover overlapping needs / interests with people in your local community and do them together!

Batch orders of supplies, picking up quick groceries for others when you're at the store, help with house repairs, etc.

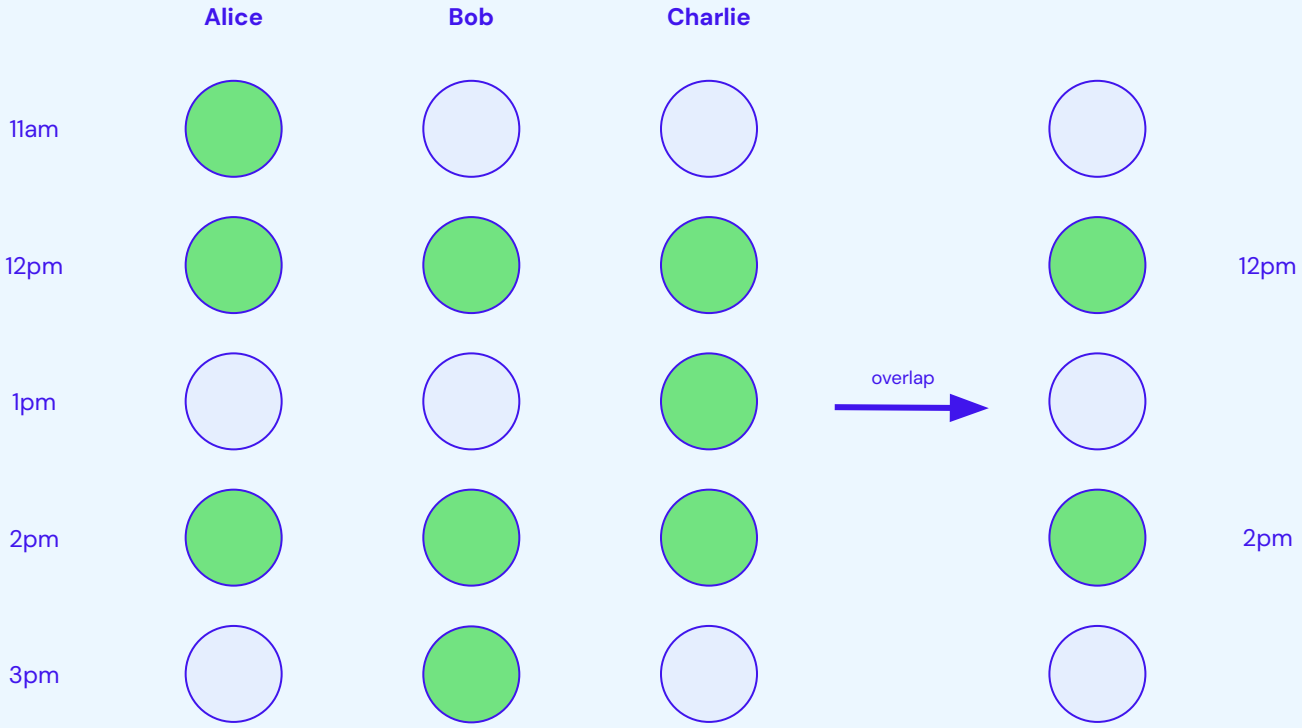
What sorts of tools do we need for digital pheromones?

(1) Private set intersection

Discover commonalities safely



Use **MPC** to find overlap without revealing anything else.



Can include more parties to **coordinate effectively**.

Reflections

- PSI uses **privacy offensively**
  - Can share your personal data safely, knowing you'll only surface commonalities
  - **Sharing maximally** vs. sharing minimally with ZK

Reflections

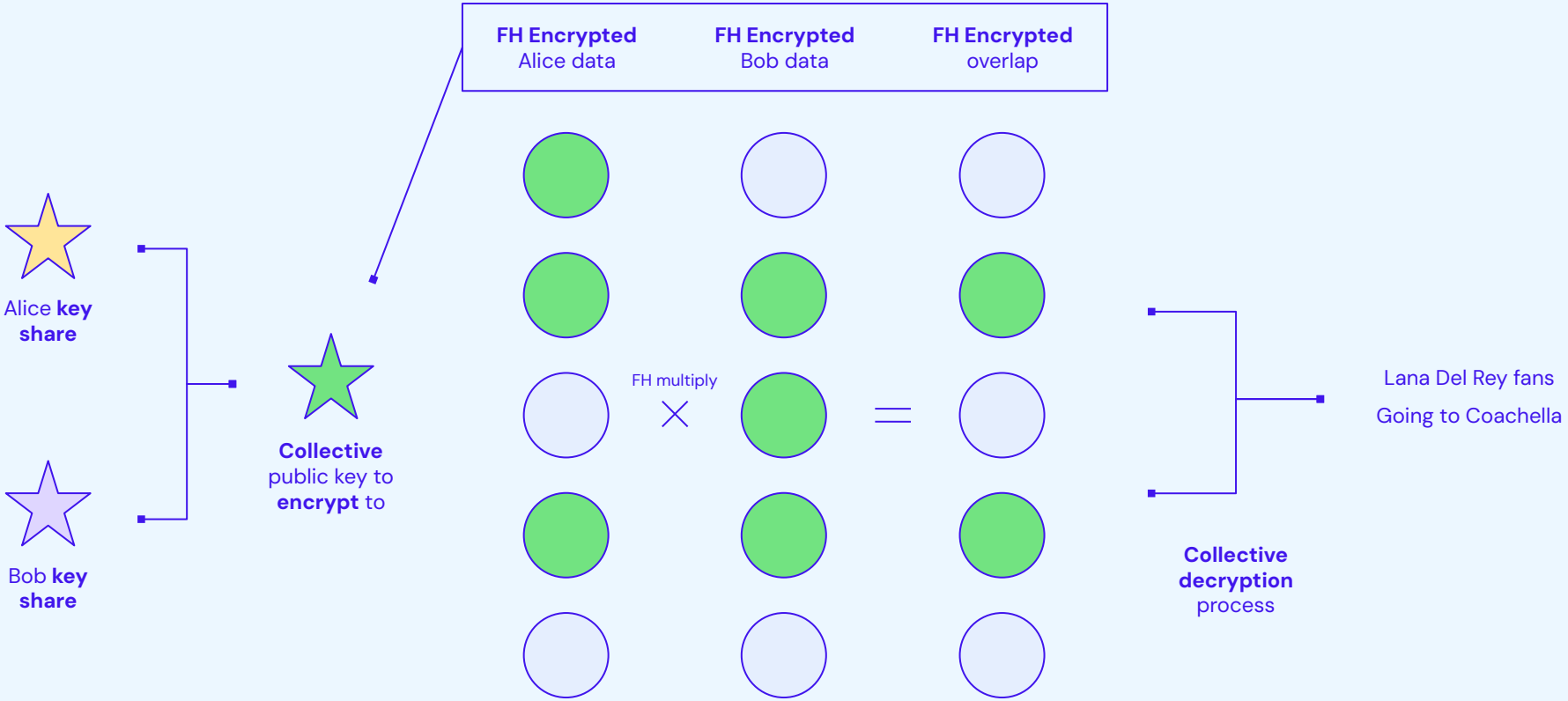
- PSI uses **privacy offensively**
  - Can share your personal data safely, knowing you'll only surface commonalities
  - **Sharing maximally** vs. sharing minimally with ZK
- PSI is very **easy to understand & explain**

Reflections

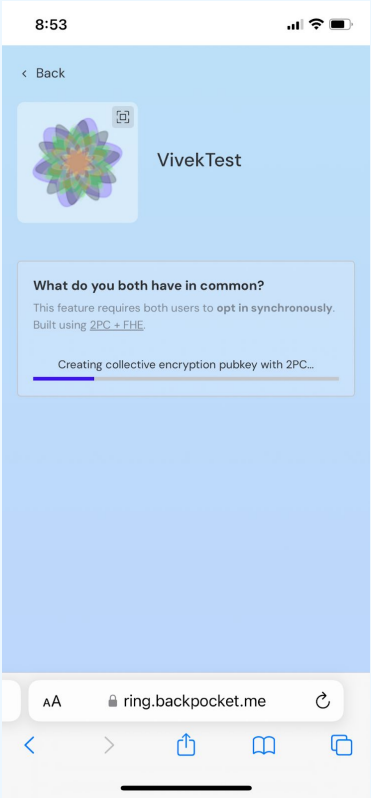
- PSI uses **privacy offensively**
  - Can share your personal data safely, knowing you'll only surface commonalities
  - **Sharing maximally** vs. sharing minimally with ZK
- PSI is very **easy to understand & explain**
- PSI is sometimes **too rigid**
  - Not just about having most in common, want to find complementary desires / strengths
  - Sometimes want to do operations like comparisons or averages

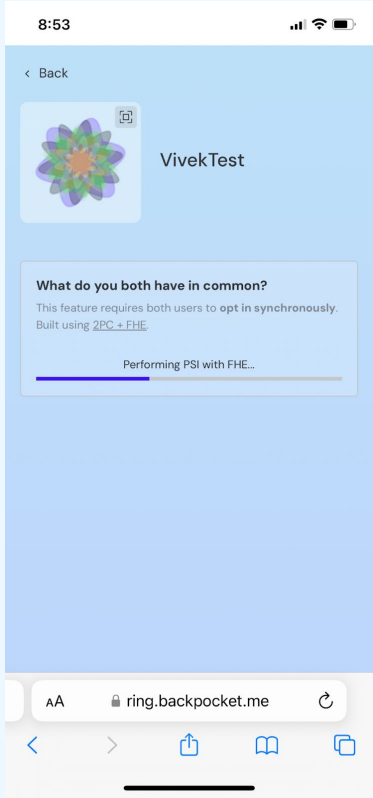
(2) Multi-party FHE

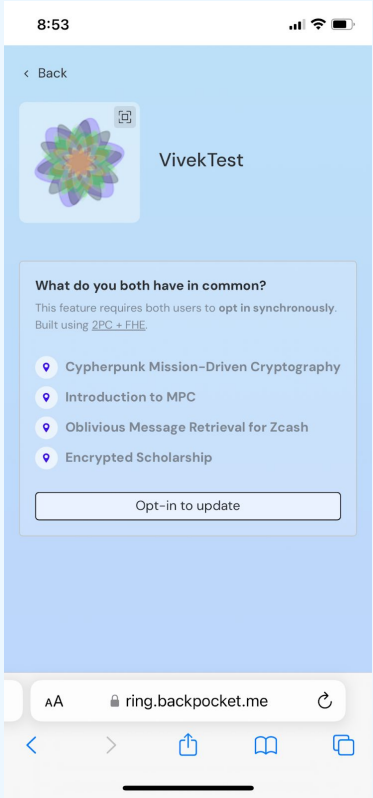
Low liveness & compute MPC



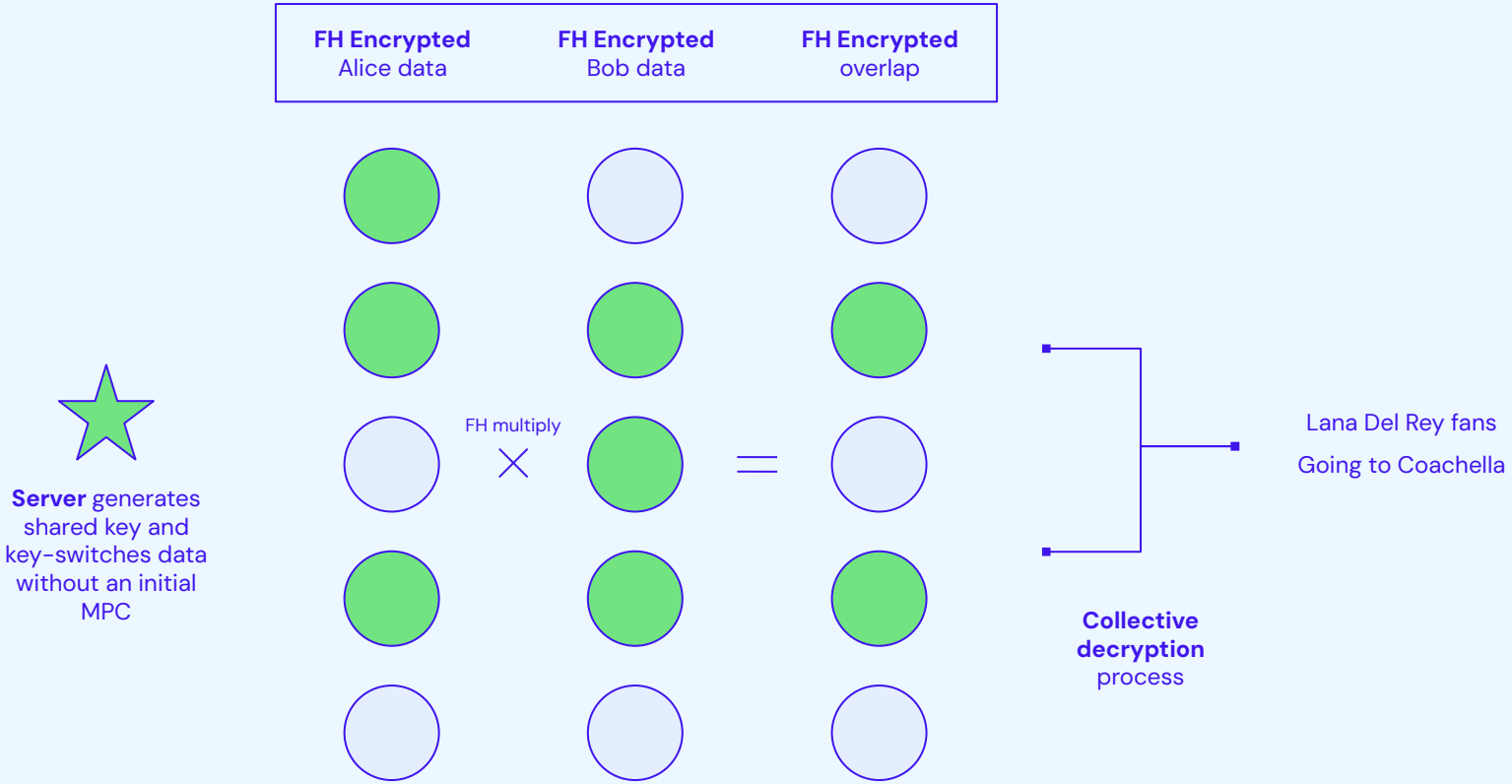
Interactive: Initial MPC for shared key + FHE compute + MPC decrypt





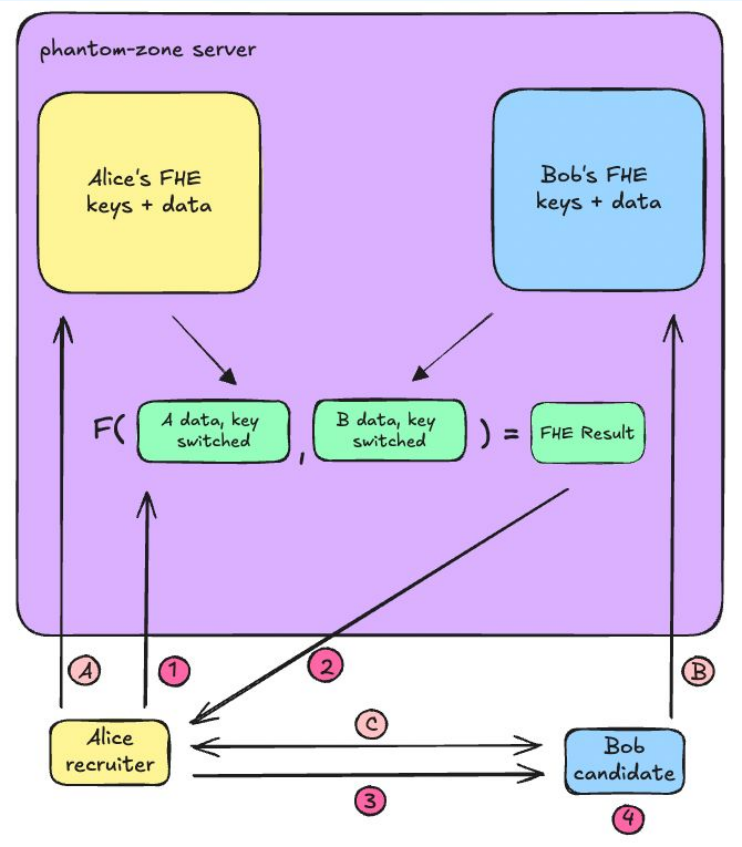


ZK Summit: PSI using Interactive Multi-Party BFV



Non-interactive: Server key-switches + FHE compute + MPC decrypt

Pioneered by Gauss Labs



Frontiers: Private job matching with Non-interactive Multi-Party FHEW

Four rounds of communication total

Reflections

- **Very little compute on-device**
  - Just generating public keys and doing encryption

Reflections

- **Very little compute on-device**
  - Just generating public keys and doing encryption
- **Nontrivial back and forth rounds**
  - Server handles creating shared key & all the compute
  - Total of 4 rounds, as decryption is in MPC

Reflections

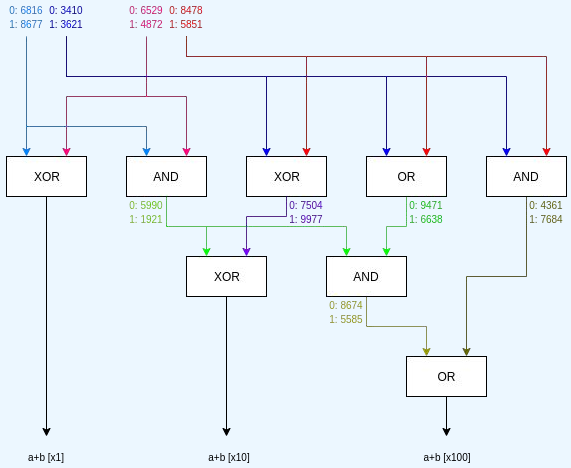
- **Very little compute on-device**
  - Just generating public keys and doing encryption
- **Nontrivial back and forth rounds**
  - Server handles creating shared key & all the compute
  - Total of 4 rounds, as decryption is in MPC
- **Uploaded public keys are huge** (15 to 100MB!)
  - Huge initial upload involved, can't function on poor Wi-Fi
  - Large encrypted data blow up as well

(3) Trinity – *new!*

Verifiable non-interactive secure computation

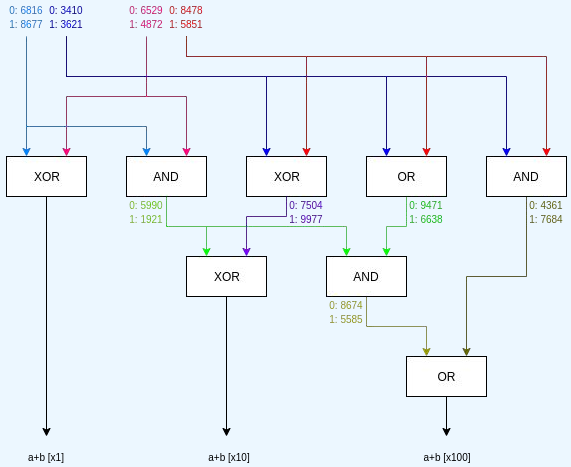
Trinity is the combination of 3 schemes

Trinity is the combination of 3 schemes



Garbled Circuits

Trinity is the combination of 3 schemes

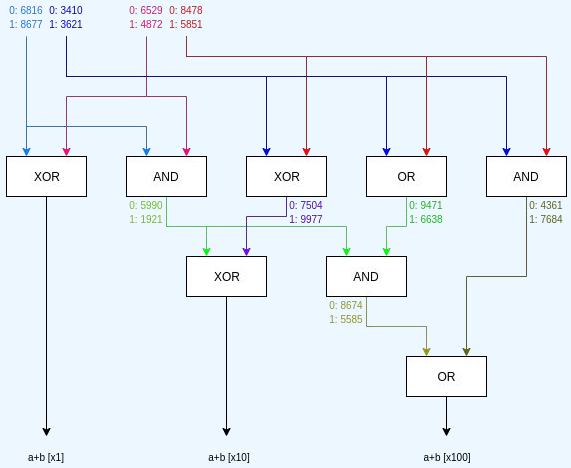


$\mathsf{Encap}^{\mathsf{H}}(\mathsf{ck}, (\mathtt{com}, \alpha, \beta))$	$\mathsf{Decap}^{\mathsf{H}}(\mathsf{ck}, (\pi_1, \ldots, \pi_\ell), \mathtt{ct})$
for $1 \leq j \leq \ell$	**parse** $\mathtt{ct}$ as $(\mathtt{ct}_1, \ldots, \mathtt{ct}_\ell)$
$\quad r_j \leftarrow \mathbb{F}_p$	**for** $1 \leq j \leq \ell$
$\quad s_j := e(r_j \cdot (\mathtt{com} - [\beta_j]_1), [1]_2)$	$\quad s_j := e(\pi_j, \mathtt{ct}_j)$
$\quad \mathtt{ct}_j \leftarrow r_j \cdot ([\tau]_2 - [\alpha_j]_2)$	$\mathtt{k} := \mathsf{H}(s_1, \ldots, s_\ell)$
$\mathtt{ct} := (\mathtt{ct}_1, \ldots, \mathtt{ct}_\ell)$	**return** $\mathtt{k}$
$\mathtt{k} := \mathsf{H}(s_1, \ldots, s_\ell)$	
return $(\mathtt{ct}, \mathtt{k})$	

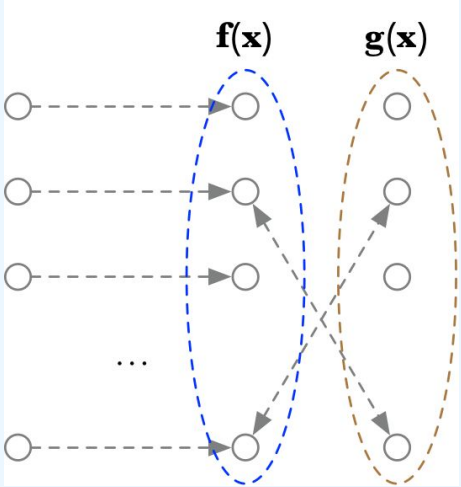
Garbled Circuits

KZG Witness Encryption

Trinity is the combination of 3 schemes



$\mathsf{Encap}^{\mathsf{H}}(\mathsf{ck}, (\mathsf{com}, \alpha, \beta))$	$\mathsf{Decap}^{\mathsf{H}}(\mathsf{ck}, (\pi_1, \ldots, \pi_\ell), \mathsf{ct})$
for $1 \le j \le \ell$	**parse** ct as $(\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$
$\quad r_j \leftarrow \mathbb{F}_p$	**for** $1 \le j \le \ell$
$\quad s_j := e(r_j \cdot (\mathsf{com} - [\beta_j]_1), [1]_2)$	$\quad s_j := e(\pi_j, \mathsf{ct}_j)$
$\quad \mathsf{ct}_j \leftarrow r_j \cdot ([\tau]_2 - [\alpha_j]_2)$	$\mathsf{k} := \mathsf{H}(s_1, \ldots, s_\ell)$
$\mathsf{ct} := (\mathsf{ct}_1, \ldots, \mathsf{ct}_\ell)$	**return** k
$\mathsf{k} := \mathsf{H}(s_1, \ldots, s_\ell)$	
return $(\mathsf{ct}, \mathsf{k})$	

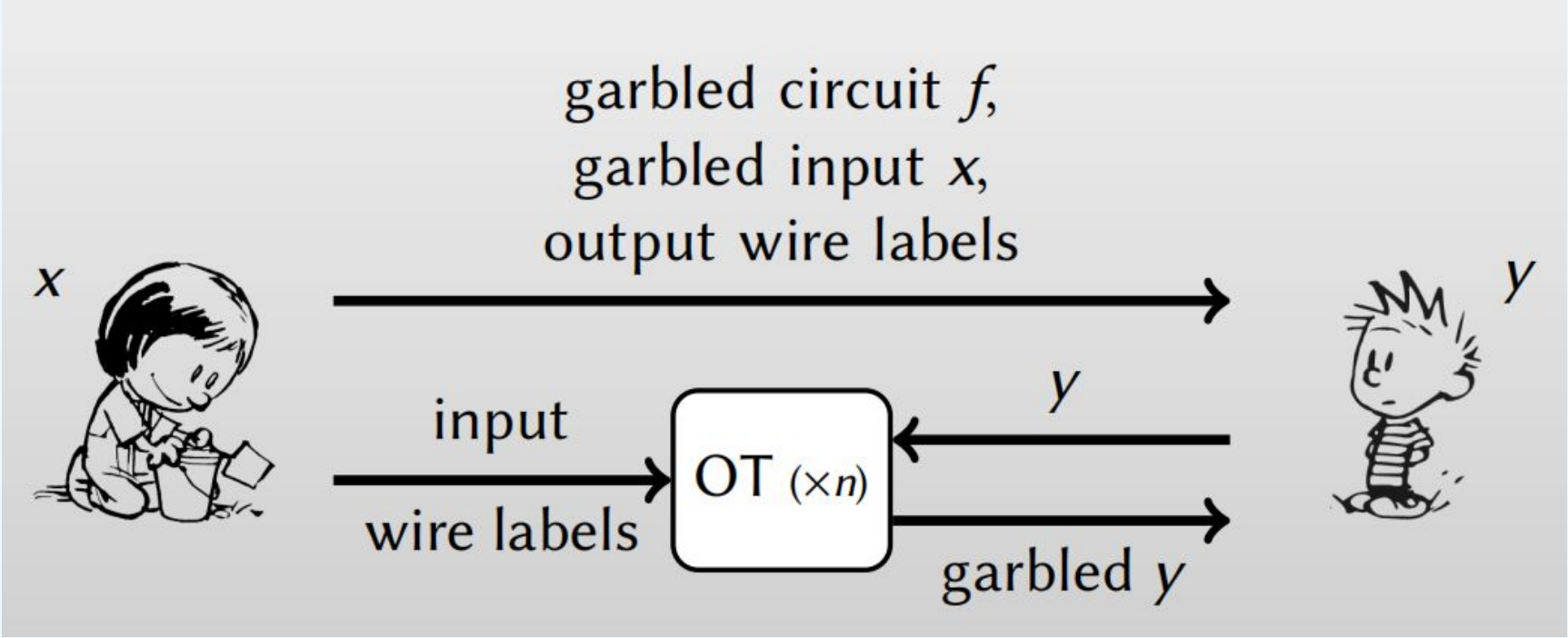


Garbled Circuits

KZG Witness Encryption

PLONK zkSNARK

Secure 2PC: Garbled Circuits



Lowering rounds: KZG Witness Encryption

$$\mathtt{ct}_{D[i]} \leftarrow \mathsf{WE.Enc}(\mathtt{pp}, (\mathtt{digest}, i, D[i]), m_{D[i]})$$

- Commit to a dictionary *D[i]* for *i = 1 to n* as a KZG commitment `digest`
- $\mathtt{ct}_{D[i]}$ is an encryption of $m_{D[i]}$ to a valid opening of `digest` at *(i, D[i])*
- If you don't have an opening proof, you can't decrypt

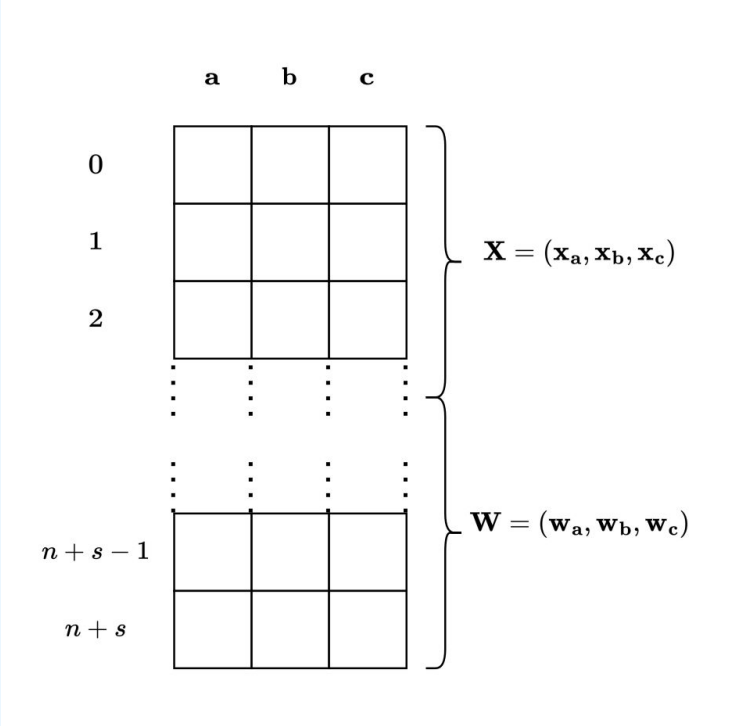
Lowering rounds: Laconic Oblivious Transfer

$\mathsf{Send}(\mathtt{pp}, \mathtt{digest}, i, m_0, m_1)$	$\mathsf{Receive}(\mathtt{pp}, \mathtt{aux}, (\mathtt{ct}_0, \mathtt{ct}_1), i)$
$\mathtt{ct}_0 \leftarrow \mathsf{WE.Enc}(\mathtt{pp}, (\mathtt{digest}, i, 0), m_0)$	$\textbf{parse}\ \mathtt{aux}\ \text{as}\ (D, \pi_1, \ldots, \pi_n))$
$\mathtt{ct}_1 \leftarrow \mathsf{WE.Enc}(\mathtt{pp}, (\mathtt{digest}, i, 1), m_1)$	$b := D_i$
$\textbf{return}\ (\mathtt{ct}_0, \mathtt{ct}_1)$	$m_b \leftarrow \underline{\mathsf{WE.Dec}}(\mathtt{pp}, \pi_i, \mathtt{ct}_b)$
	$\textbf{return}\ m_b$

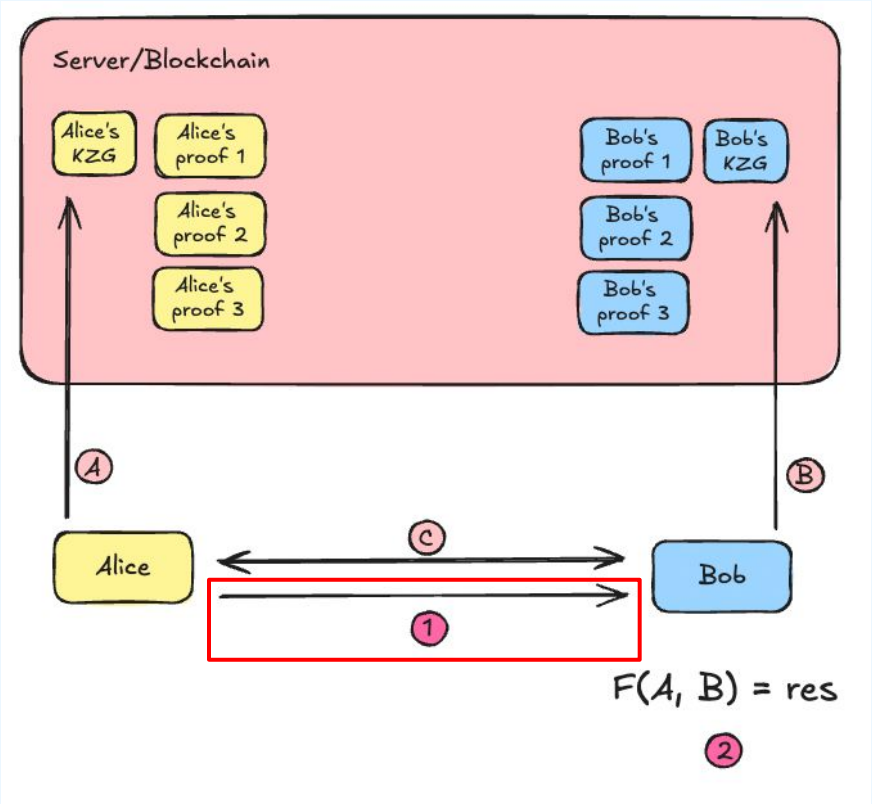
- Can use KZG to batch many OT inputs into one commitment
- Can use KZG Witness Encryption to build 1-of-2 OT on this commitment
- Can send the OT data alongside the garbling, 1 round total

Verifiable inputs: PLONK

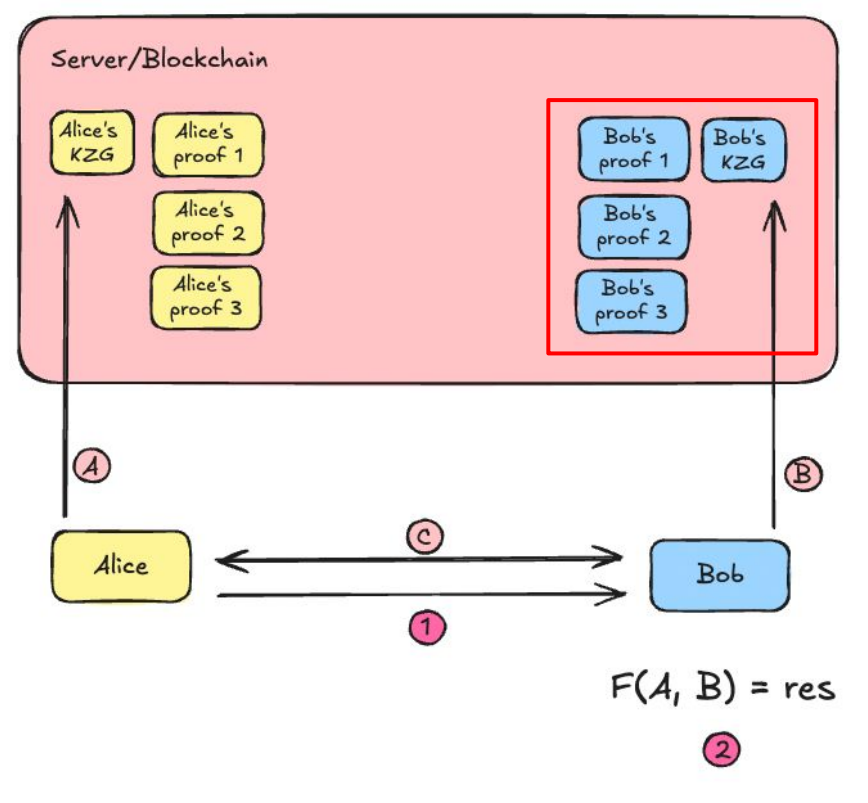
- **KZG commit garbled circuit inputs** to use Laconic OT
- Can make KZG commitment first column in a PLONK proof
- Use other columns to **prove the commitment is "valid"** using zkEmail / zkTLS data



Trinity needs *one round of data transfer* for 2PC



Trinity has *succinct validity proofs of Bob's inputs*



Reflections

- Trinity enables very **simple DevX and UX** for consumer 2PC
  - Minimizing rounds is very important as phones get turned off

Reflections

- Trinity enables very **simple DevX and UX** for consumer 2PC
  - Minimizing rounds is very important as phones get turned off
- Trinity can build off **other ZK-proven data** about users

Reflections

- Trinity enables very **simple DevX and UX** for consumer 2PC
  - Minimizing rounds is very important as phones get turned off
- Trinity can build off **other ZK-proven data** about users
- Overall: **Send an encrypted email, only decrypt if you match criteria**
  - Send a job description + requirements, candidate only sees details if they fit criteria
  - Send a dating profile, matches only see details if they fit your criteria
  - Invite people to CO-LAB side event, only see details if they like coSNARKs

Counterpoint: TEEs

Not neutral or peer-to-peer!

Where can I experience this technology?

Cursive Connections (more data coming soon!)

Ethereum hot takes

zero-knowledge proofs

Based | Neutral | Cringe

Devcon events

- Opening Ceremony
- Why VPNs are Scams and what to do about it
- Crypto is the Real World: Understanding the Crypto...
- EIPs Simplified: History and Process Explained

9:41

connections.cursive.team

Back

Alice
@alice2000

Add Notes | Devcon SEA | Edge City Lanna

Socials

Telegram — @alice

X — @aalice00

Discover intersections in your encrypted data.

Watch your shared flower grow the more you learn about one another!

Creating collective encryption pubkey with 2PC...

Github

Username — @alice2000

Foundry
1st commit: 1990 Total: 1.5k Rank: 2

Rust
1st commit: 1990 Total: 1.5k Rank: 2

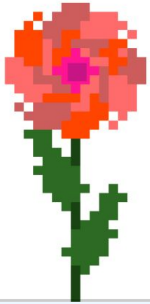
Shared availability

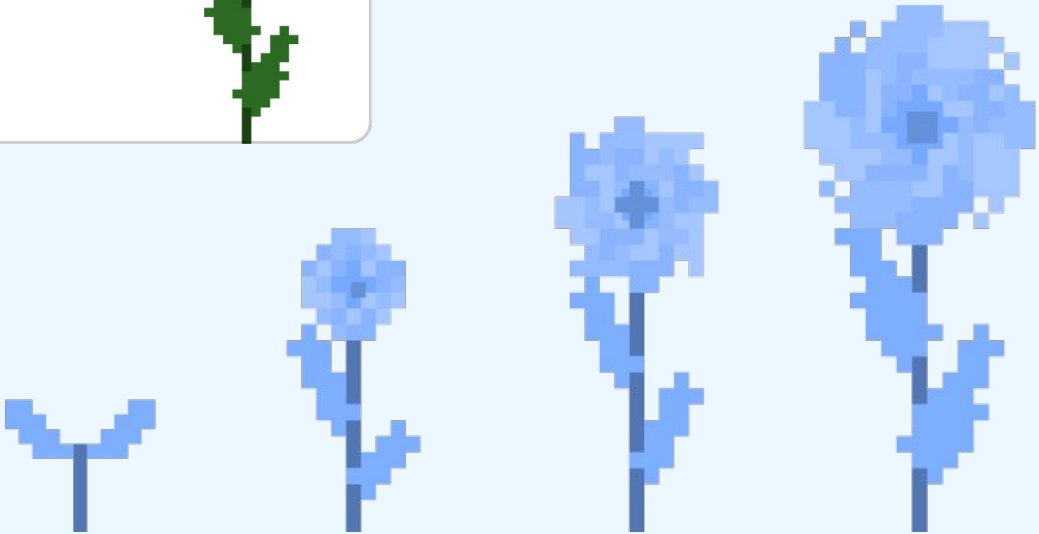
Nov 12	08:00–13:00	14:30–16:30	17:00–18:30
Nov 13	08:00–13:00	14:30–16:30	17:00–18:30
Nov 14	08:00–13:00	14:30–16:30	17:00–18:30
Nov 15	08:00–13:00	14:30–16:30	17:00–18:30

Grow your digital flower garden!

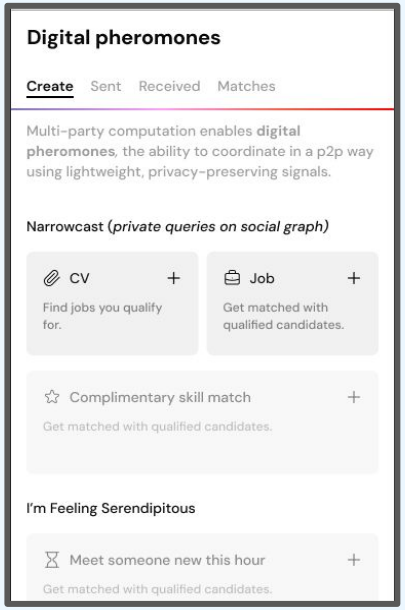
Discover intersections in your encrypted data.

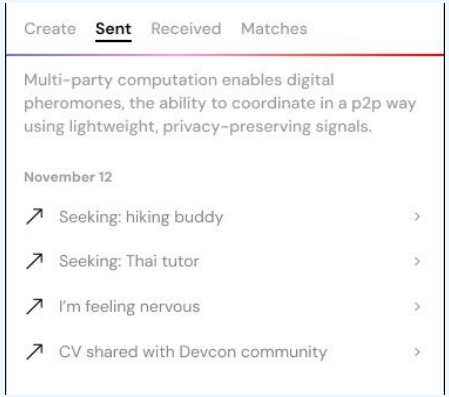
Watch your shared flower grow the more you learn about one another!

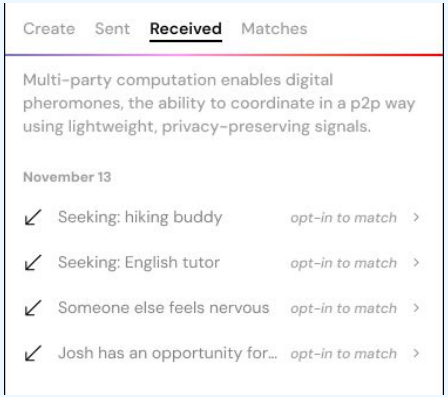


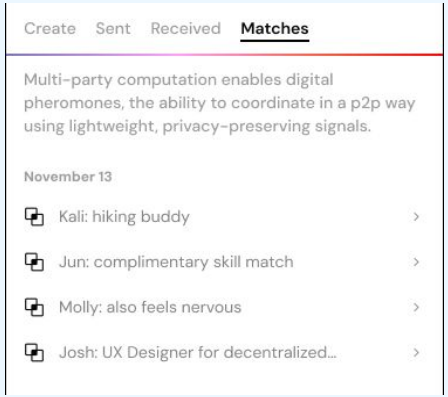


Soon: Digital pheromones + Cursive social graph

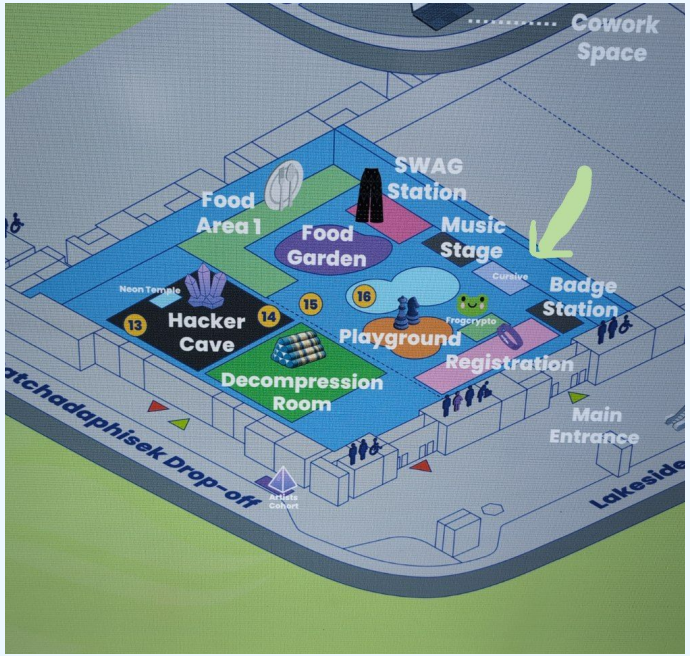






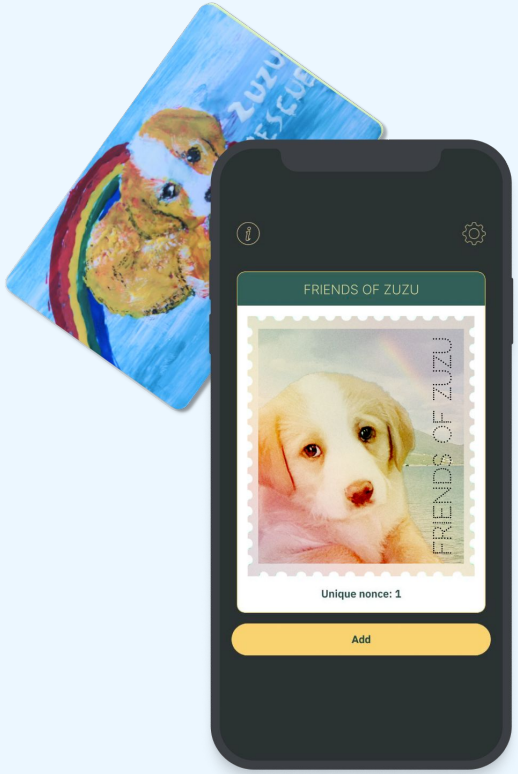


Cryptographic Connections Museum!



Cryptographic Classroom

- Explaining our recent research over a blackboard
- Going through classic cryptography constructions (KZG, Garbled Circuits, PLONK, TFHE)
- **Thursday & Friday 10:30am – 12pm** (maybe more sessions!)



Cursive is a research & design lab building cryptography for human connection.

privacy + scaling explorations

@cursive_team
Twitter

Cursive
Telegram