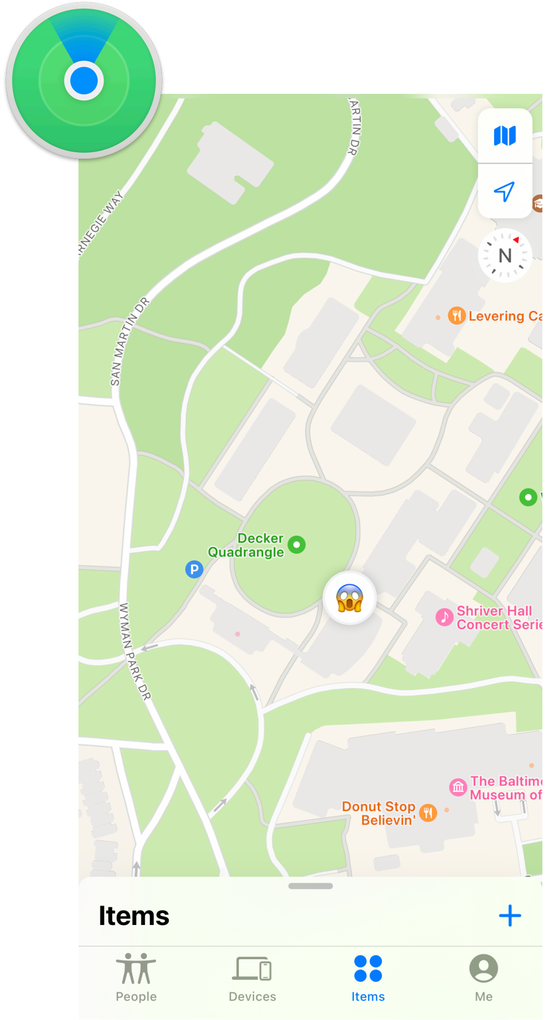
Who tracks the trackers?

Balancing privacy and stalker detection for Apple's AirTags

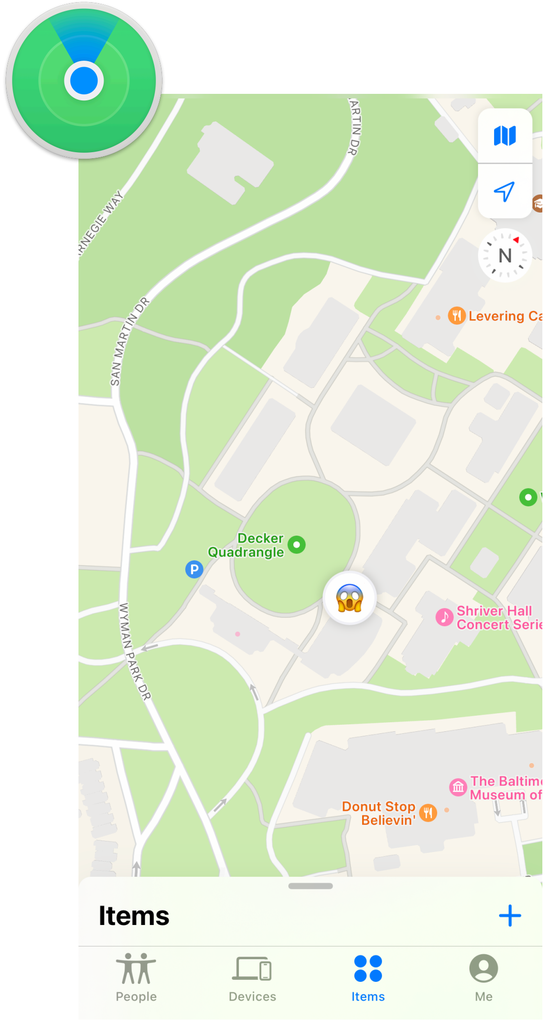
Talk based on "*Abuse-Resistant Location Tracking: Balancing Privacy and Safety in the Offline Finding Ecosystem" by* ***Gabrielle Beck****, Harry Eldridge, Matthew Green, Nadia Heninger, and Abhishek Jain. https://eprint.iacr.org/2023/1332.pdf*

The AirTag


ARTIN DR
NEGIE WAY
SAN MARTIN DR
N
Levering Ca
Decker
Quadrangle
P
Shriver Hall
Concert Serie
WYMAN PARK DR
The Baltim
Museum of
Donut Stop
Believin'
Items
People
Devices
Items
Me

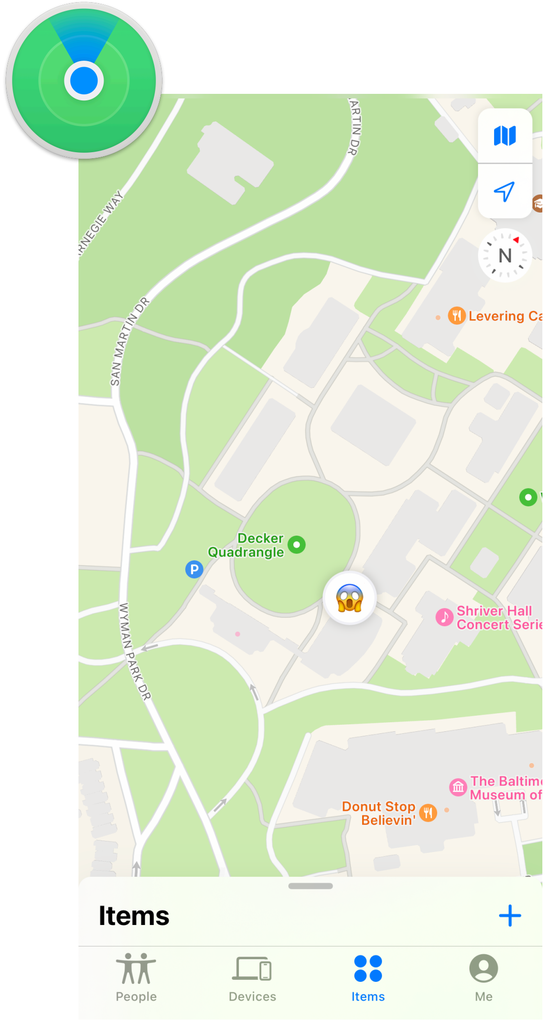
The AirTag

- Tracks physical objects: keys, luggage, pets, etc.



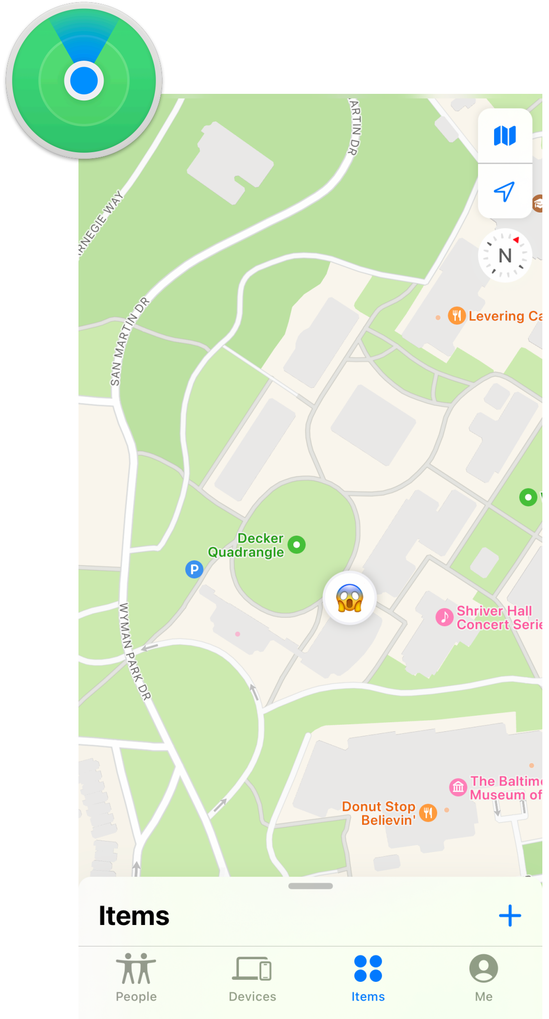
The AirTag

- Tracks physical objects: keys, luggage, pets, etc.
- Works via Find My



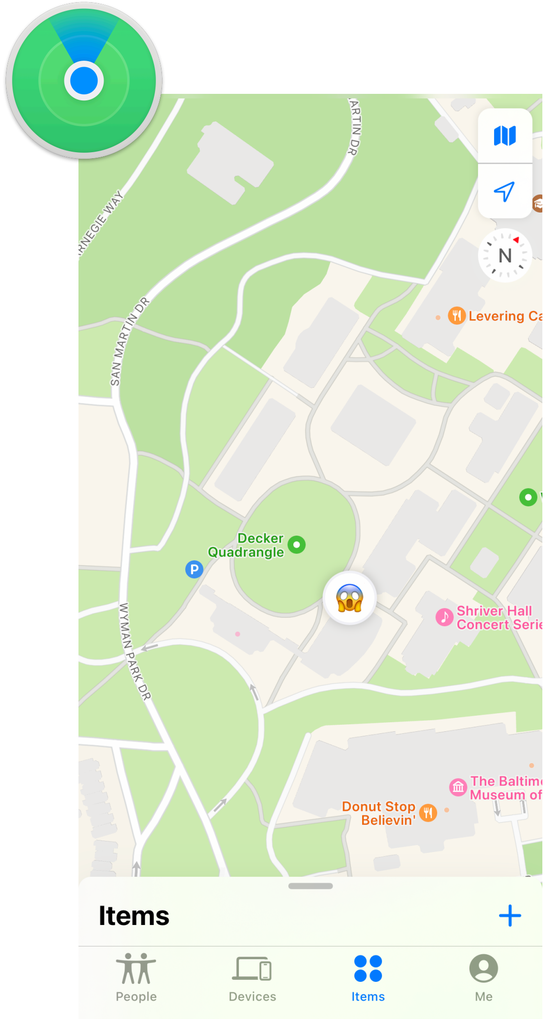
The AirTag

- Tracks physical objects: keys, luggage, pets, etc.
- Works via Find My
  - *crowd-sourced* location tracking system

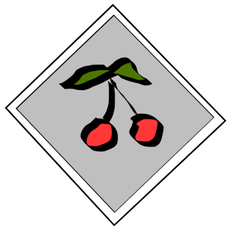


The AirTag

- Tracks physical objects: keys, luggage, pets, etc.
- Works via Find My
  - *crowd-sourced* location tracking system
  - uses no GPS, no Internet



How does it work though?

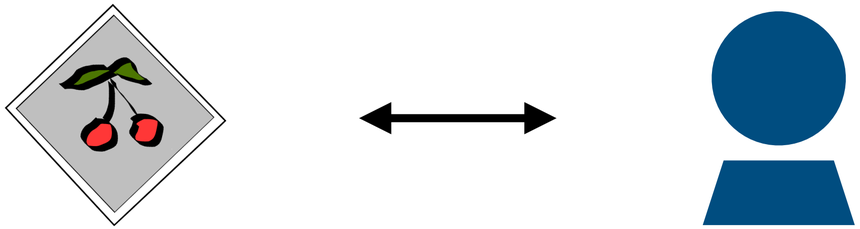


Tag

Tag Owner

How does it work though?

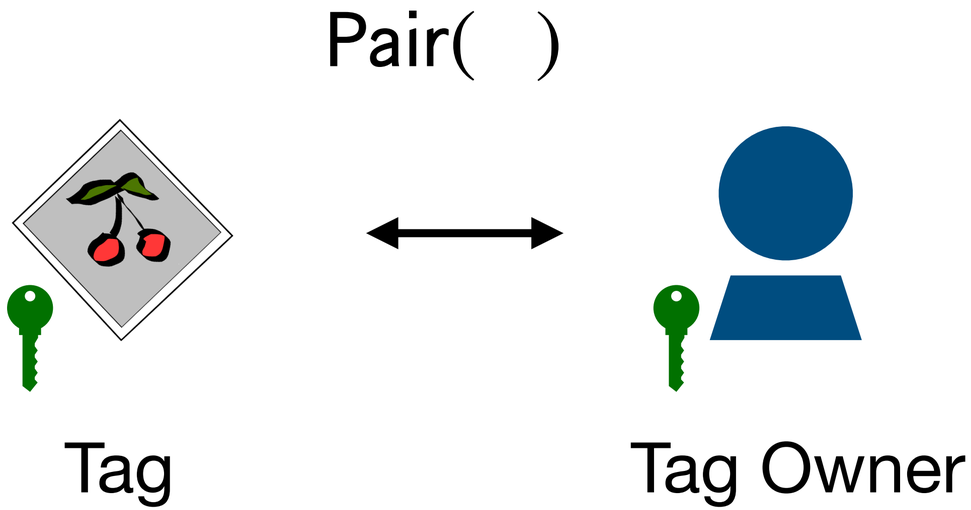
Pair( )



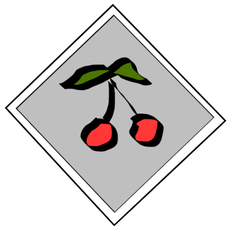
Tag

Tag Owner

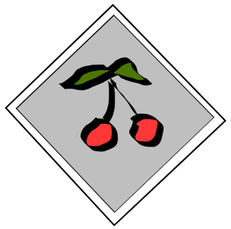
How does it work though?



How does it work though?

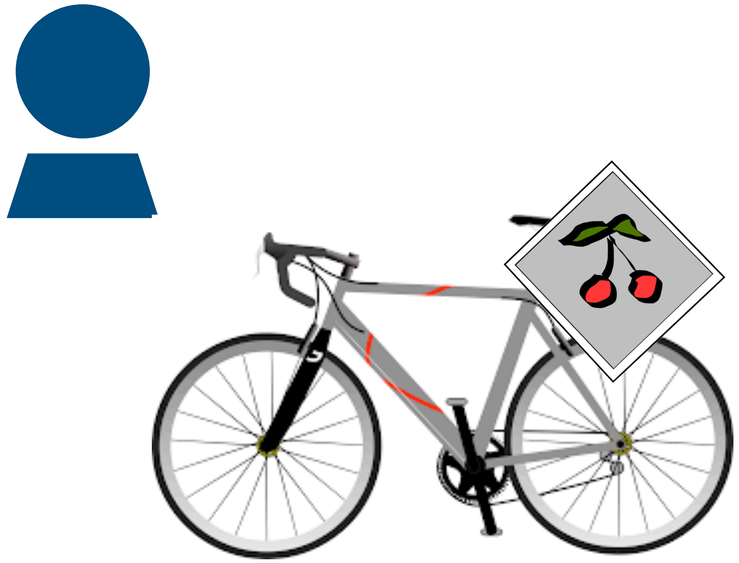


How does it work though?



How does it work though?

How does it work though?

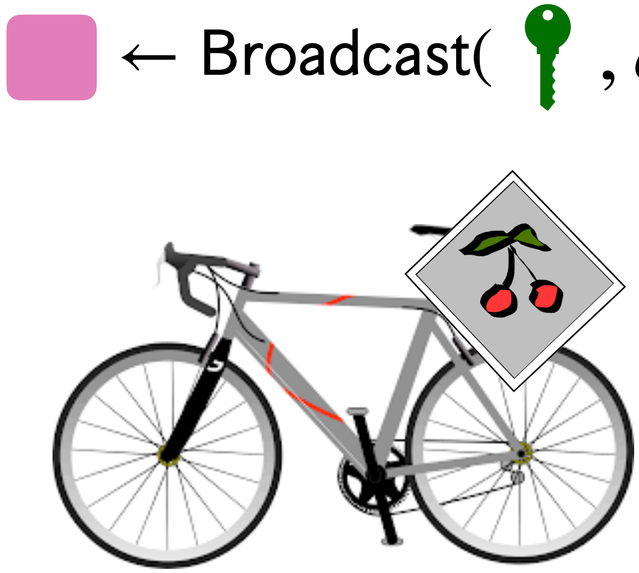


How does it work though?

How does it work though?

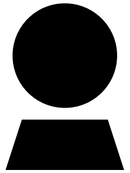
How does it work though?

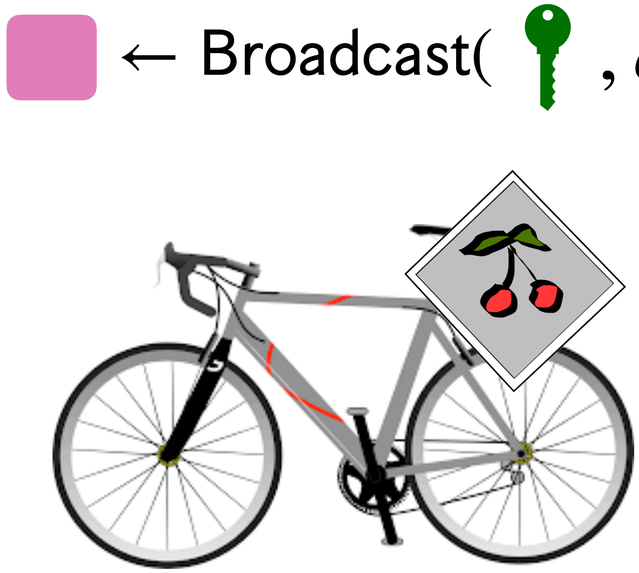
 - “volunteers”



How does it work though?

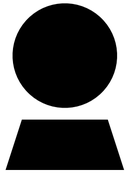
 - “volunteers”

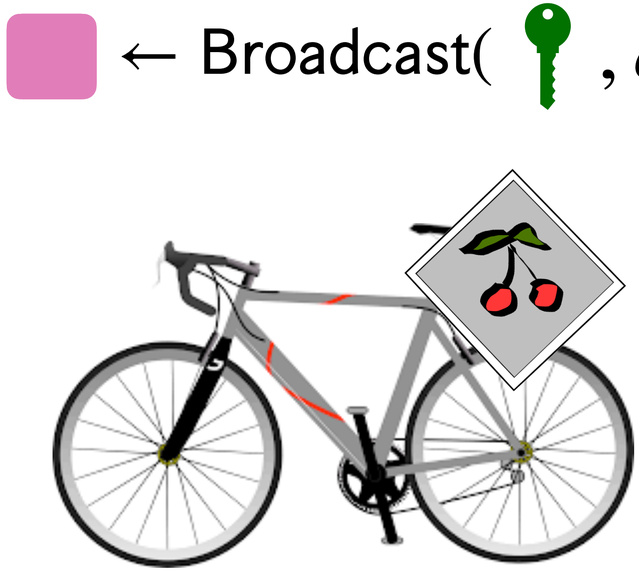


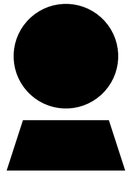


How does it work though?

 - “volunteers”

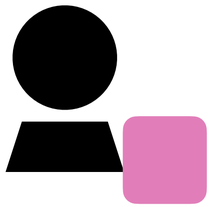


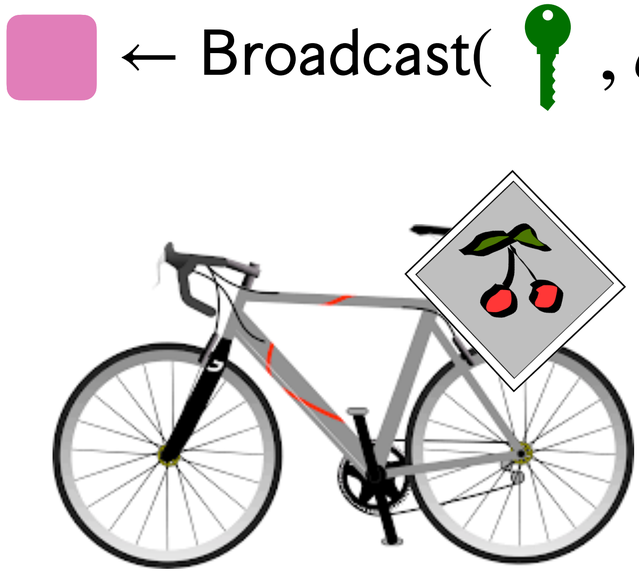


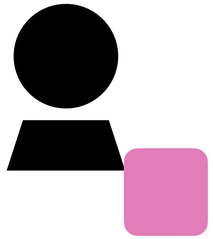


How does it work though?

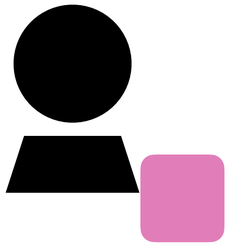
 - “volunteers”

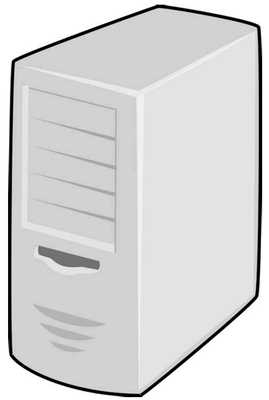






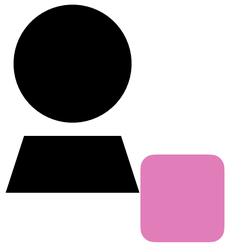
How does it work though?

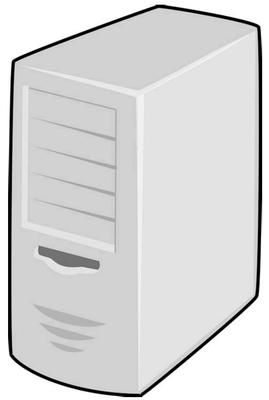




Server

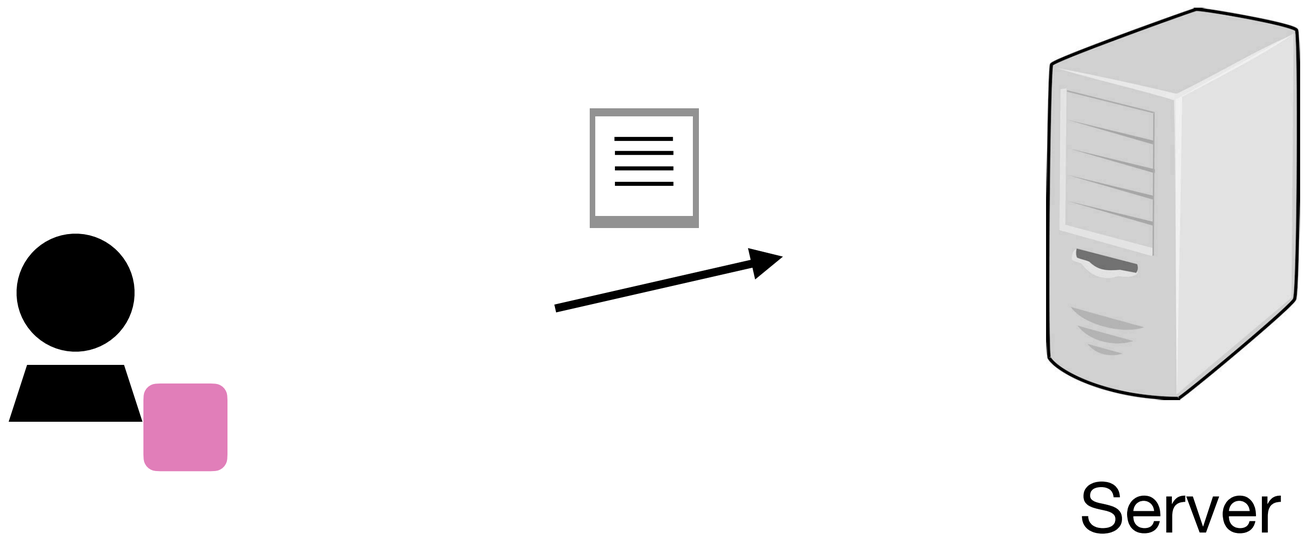
How does it work though?



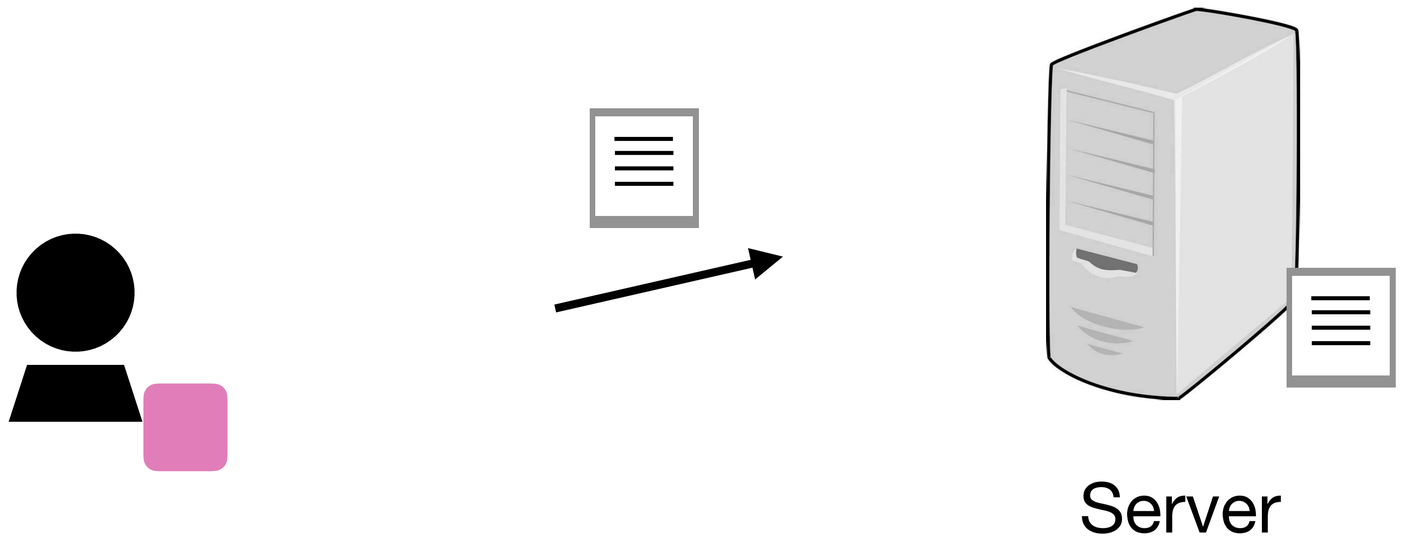


Server

How does it work though?

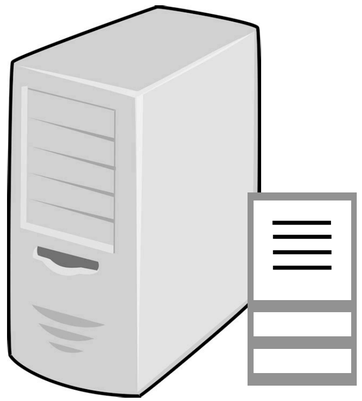


How does it work though?



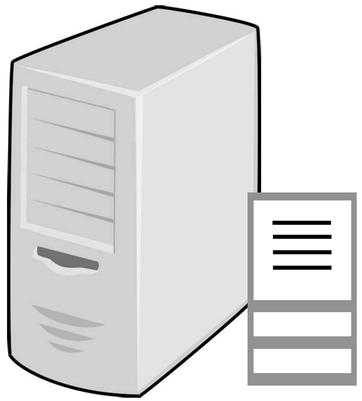
How does it work though?

Retrieve( )

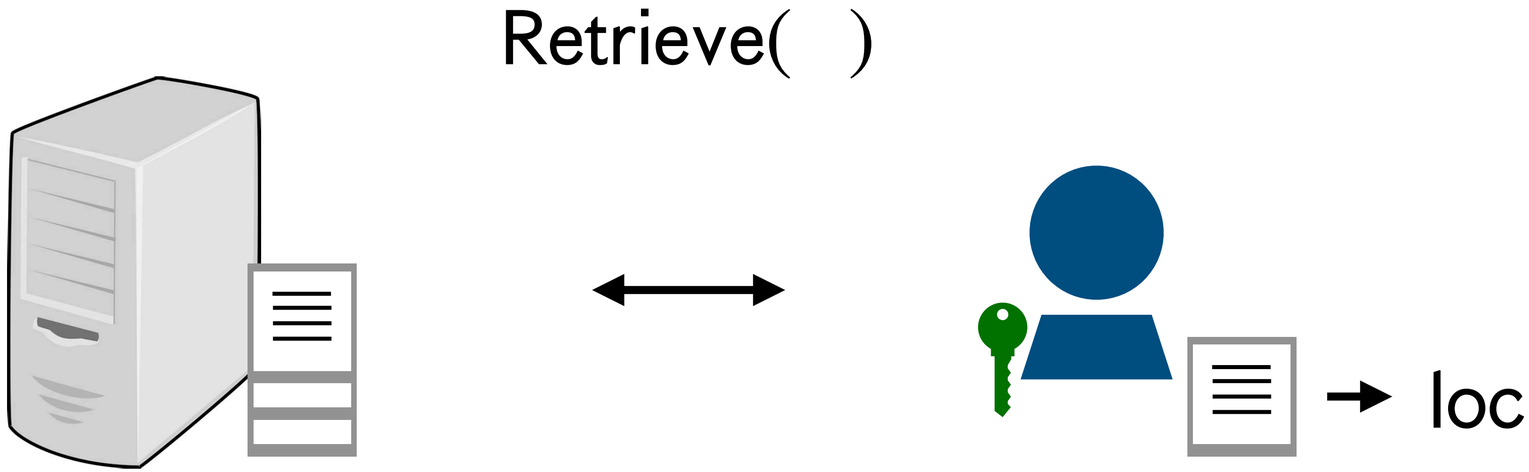


How does it work though?

Retrieve( )



How does it work though?

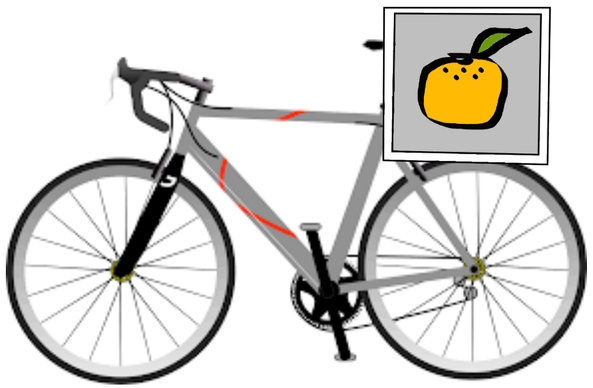


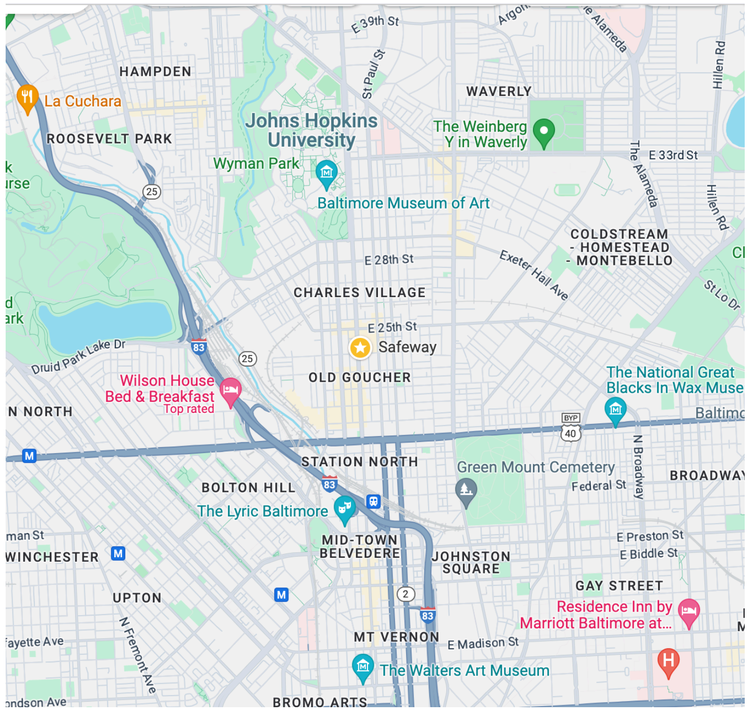
Sounds great, what could go wrong?

Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same device could **link** them together

Bxs:

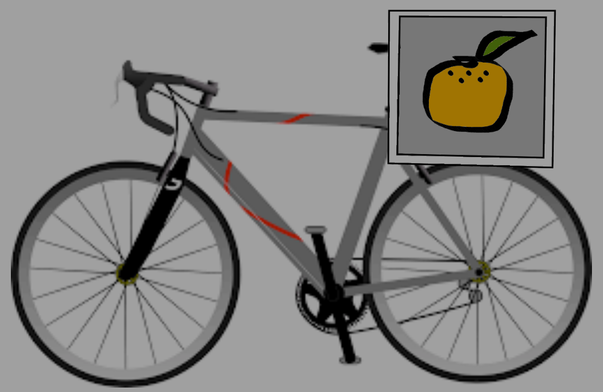


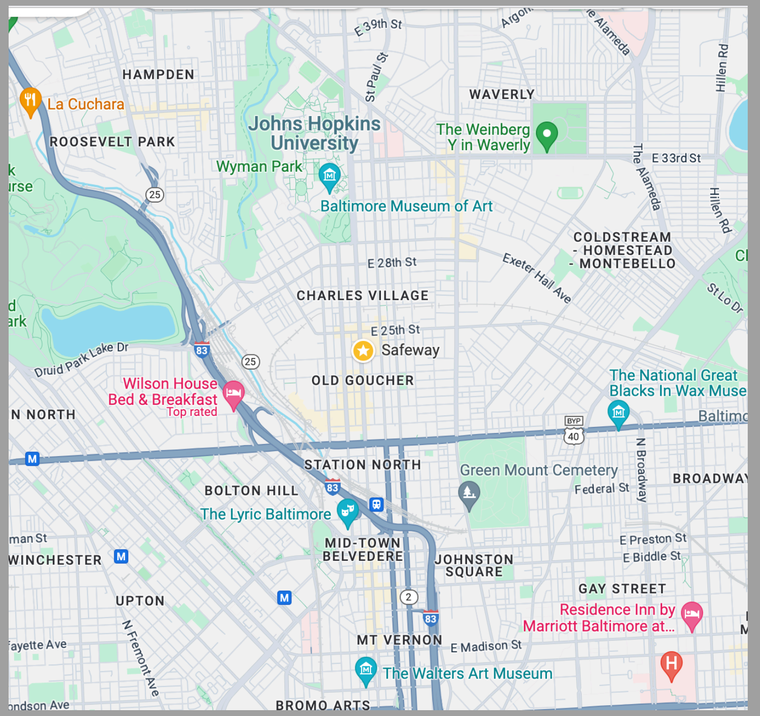


Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same device could **link** them together

Bxs:

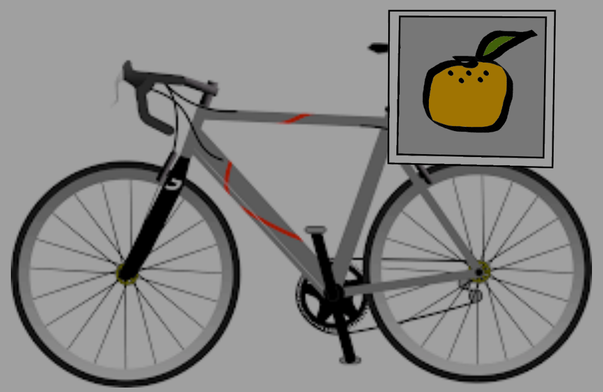


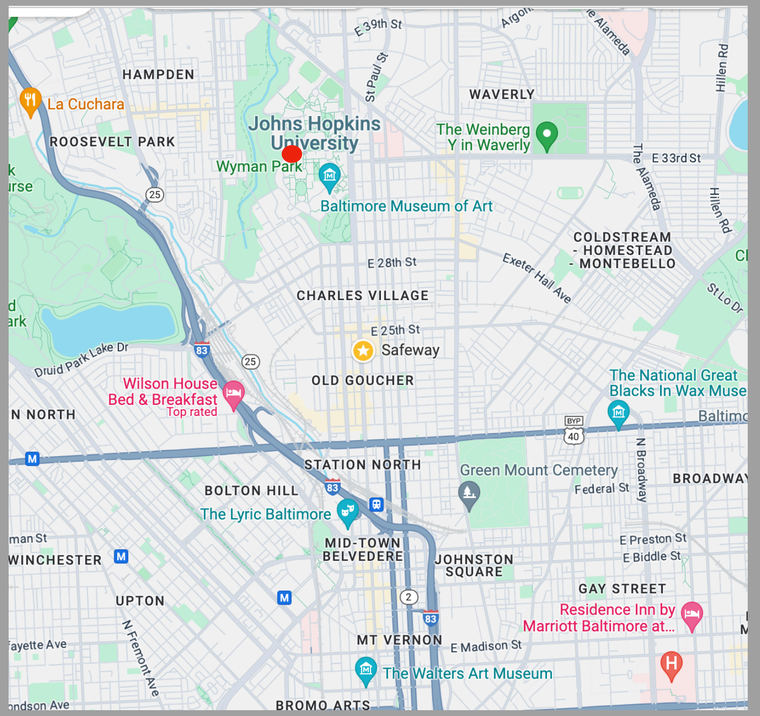


Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same device could **link** them together

Bxs:

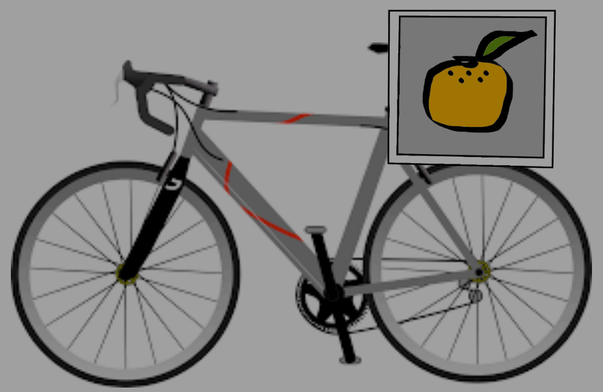


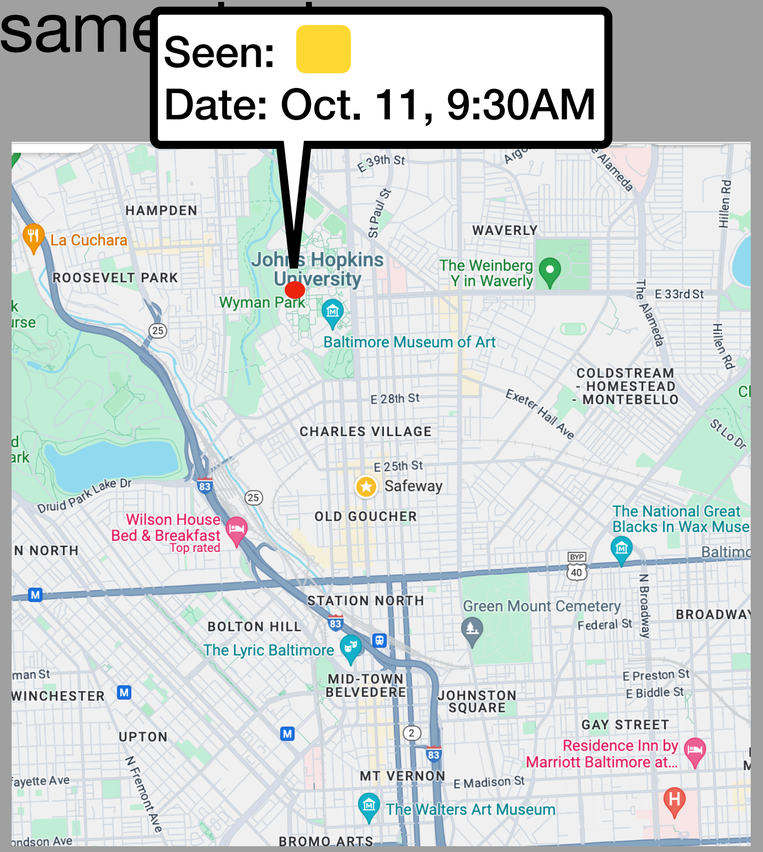


Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same [illegible] could **link** them together

Bxs:

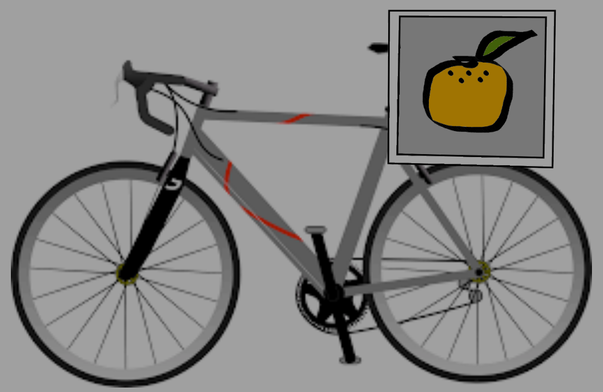


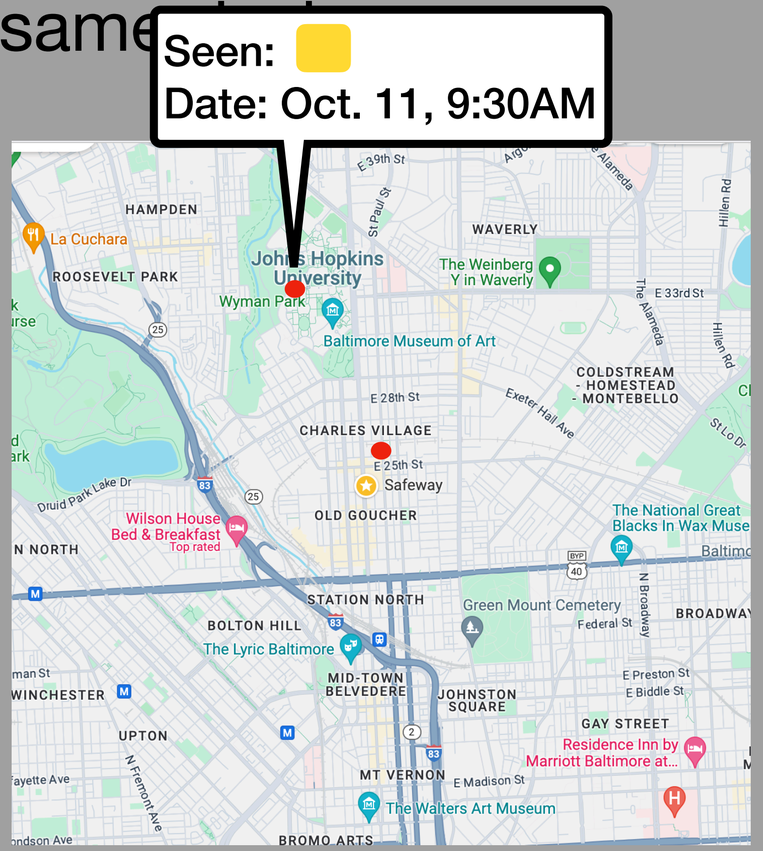


Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same [illegible] could **link** them together

Bxs:

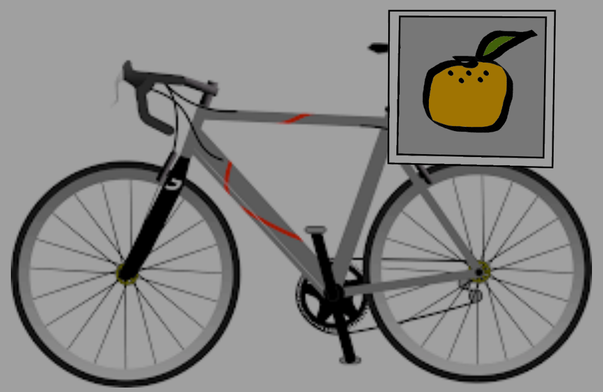


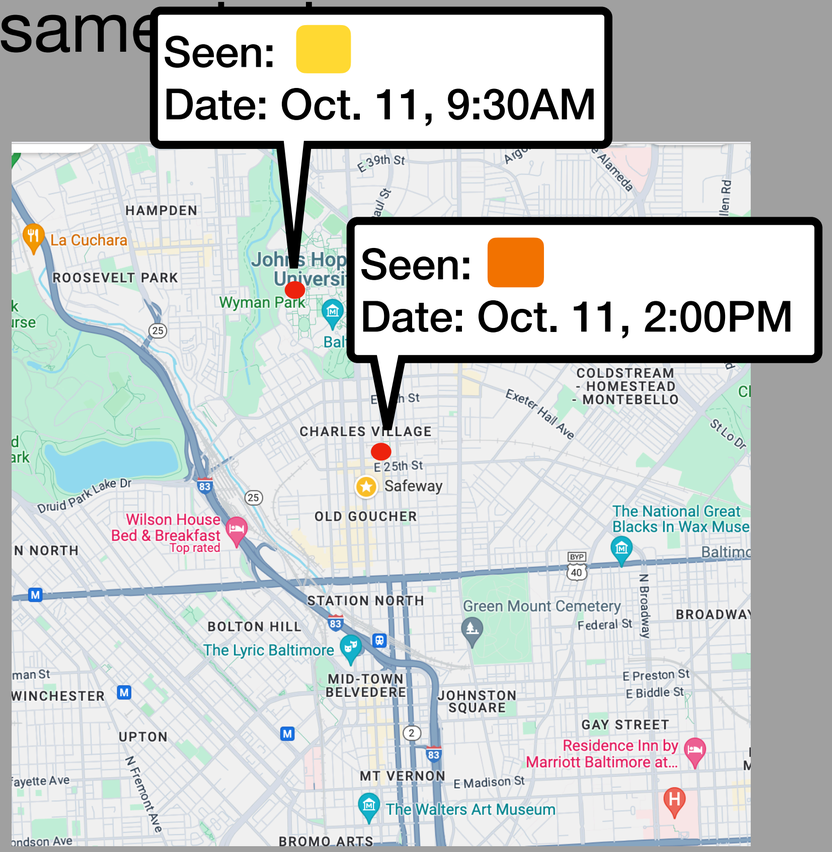


Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same [illegible]
could **link** them together

Bxs:



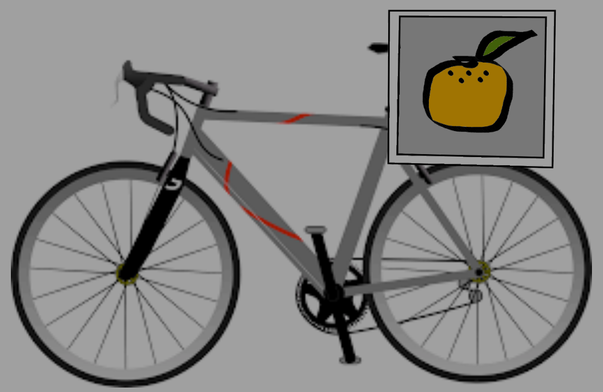


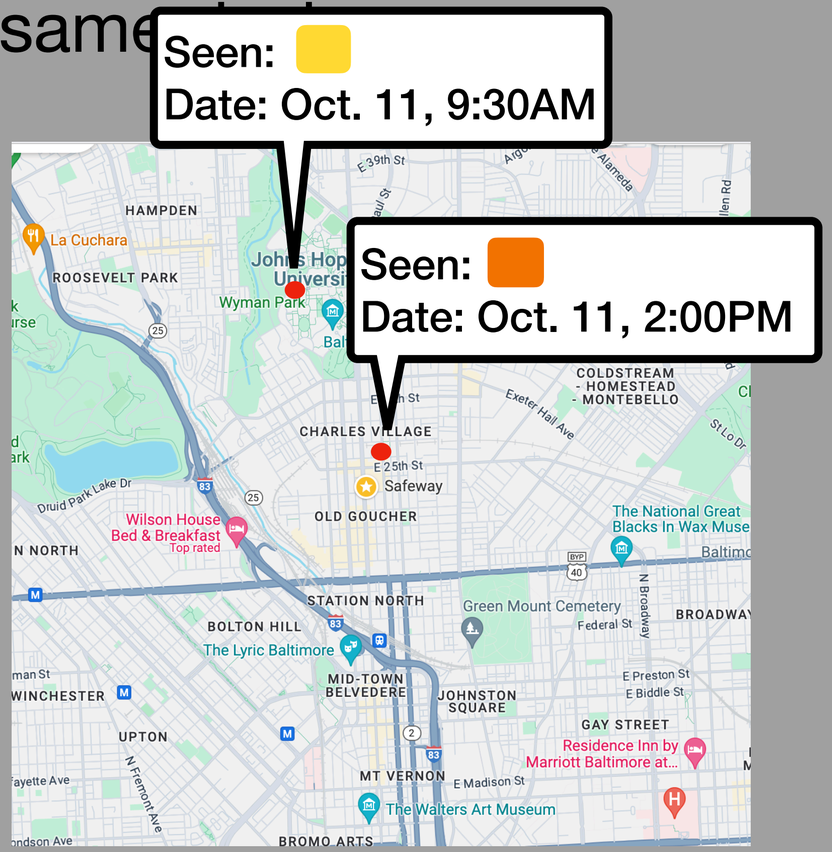
Potential Safety Risks - Tracking?

An adversary who sees multiple bxs from the same [illegible]
could [illegible]

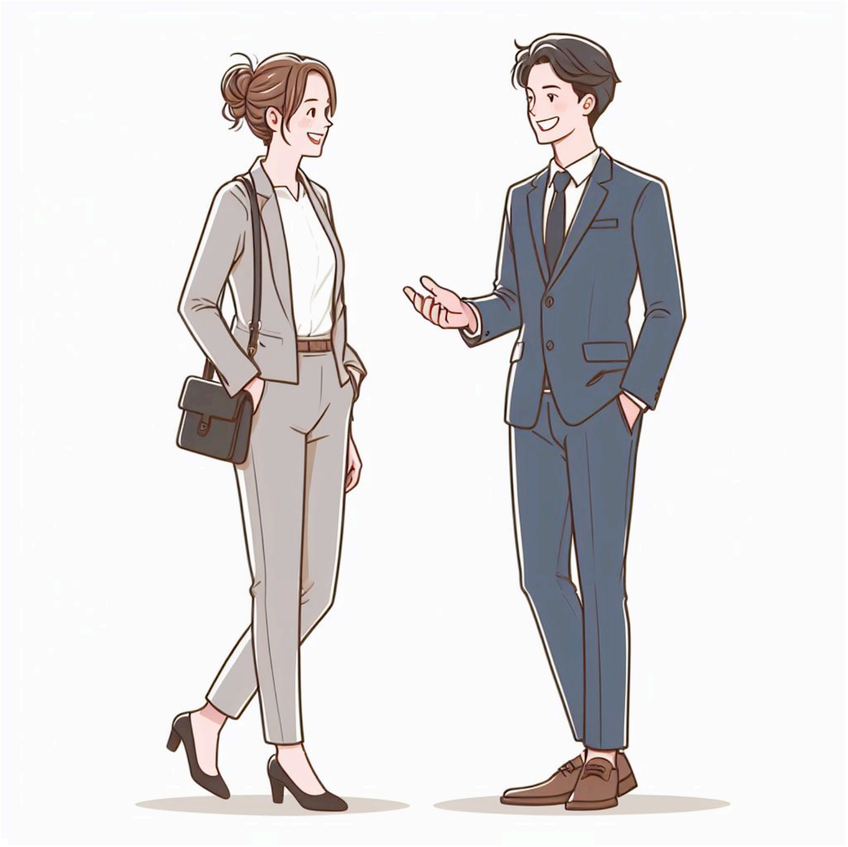
Is this the same person?

Bxs:

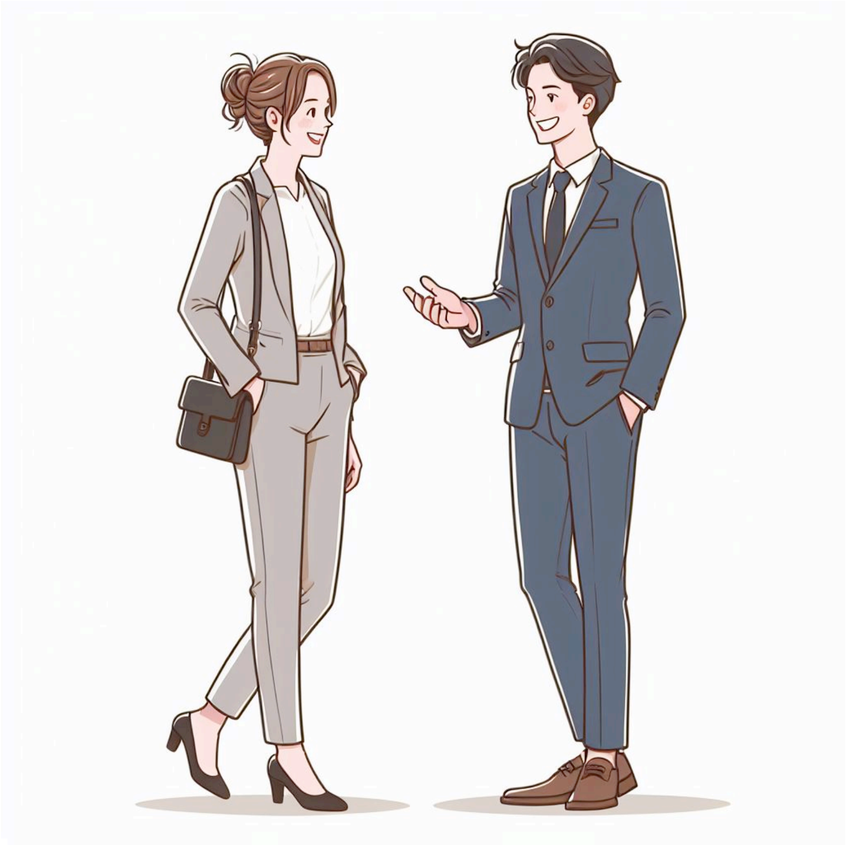




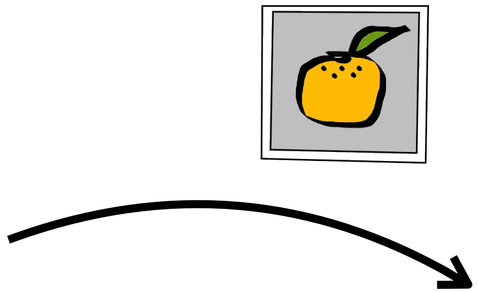
Potential Safety Risks - Stalking?

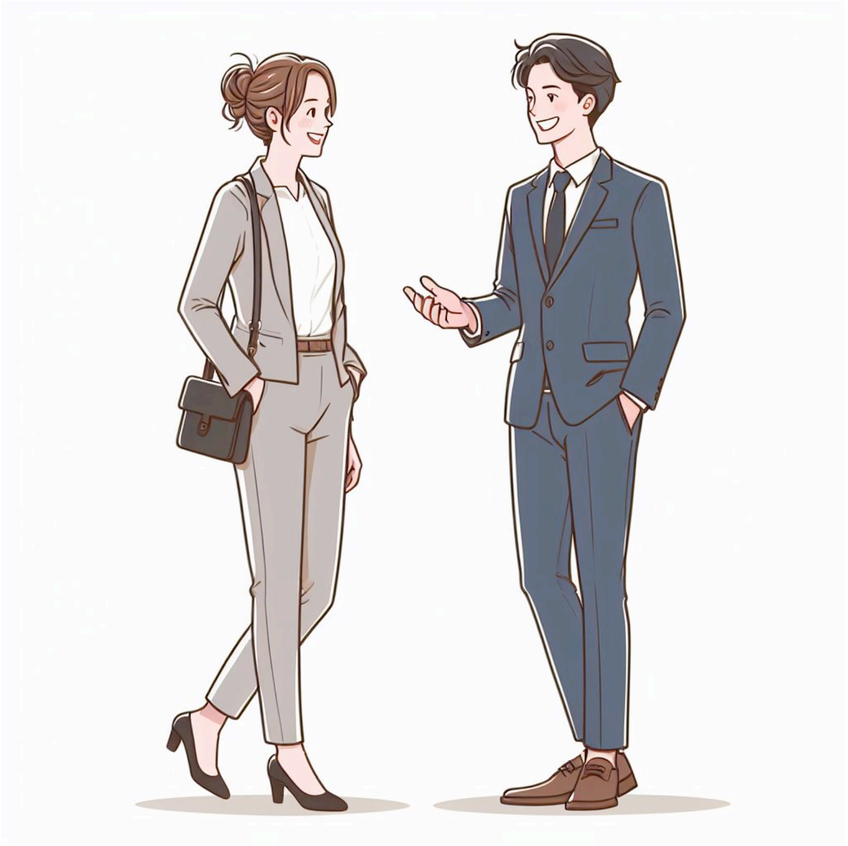


Potential Safety Risks - Stalking?



Potential Safety Risks - Stalking?





AirTag Security Properties?

Privacy is built in.

Only you can see where your AirTag is. Your location data and history are never stored on the AirTag itself. Devices that relay the location of your AirTag also stay anonymous, and that location data is encrypted every step of the way. So not even Apple knows the location of your AirTag or the identity of the device that helps find it.

AirTag is designed to discourage unwanted tracking. If someone else's AirTag finds its way into your stuff, your iPhone will notice it's traveling with you and send you an alert. After a while, if you still haven't found it, the AirTag will start playing a sound to let you know it's there.

Of course, if you happen to be with a friend who has an AirTag, or on a train with a whole bunch of people with AirTag, don't worry. These alerts are triggered only when an AirTag is separated from its owner.

Apple. https://www.apple.com/airtag/. Accessed on: Wayback Machine. April 2021

AirTag Security Properties?

- Broadcast produces **pseudorandom** value based on  and epoch e

AirTag Security Properties?

- Broadcast produces **pseudorandom** value based on  and epoch e
  - Small epoch duration = good privacy for tag user

AirTag Security Properties?

- Broadcast produces **pseudorandom** value based on  and epoch e
  - Small epoch duration = good privacy for tag user
- What about stalker detection??

AirTag Security Properties?

- Broadcast produces **pseudorandom** value based on  and epoch e
  - Small epoch duration = good privacy for tag user
- What about stalker detection??
  - … make epoch longer?

AirTag Security Properties?

- Broadcast produces **pseudorandom** value based on  and epoch e
  - Small epoch duration = good privacy for tag user
- What about stalker detection??
  - … make epoch longer?

Currently, epoch duration is **24 hrs** for separated mode

Does the implementation meet security goals?

Does the implementation meet security goals?

- Despite long tracking period, conditions for **alerting** users to stalking are **opaque** and **confusing**

Does the implementation meet security goals?

- Despite long tracking period, conditions for **alerting** users to stalking are **opaque** and **confusing**
- **Honest tag users** have **low privacy**

Does the implementation meet security goals?

- Despite long tracking period, conditions for **alerting** users to stalking are **opaque** and **confusing**
- **Honest tag users** have **low privacy**
  - Can be tracked for 24 hrs.

DULT IETF Draft

- Proposed standards for protection against unwanted trackers/stalking tags
  - Would apply to **all** trackers (e.g. Apple's, Google's, Tile's, Samsung's, etc.)

DULT IETF Draft

3.5.1. Rotation policy

An accessory **SHALL** rotate its address on any transition from near-owner state to separated state as well as any transition from separated state to near-owner state.¶

When in near-owner state, the accessory **SHALL** rotate its address every 15 minutes. This is a privacy consideration to deter tracking of the accessory by non-owners when it is in physical proximity to the owner.¶

When in a separated state, the accessory **SHALL** rotate its address every 24 hours. This duration allows a platform's unwanted tracking algorithms to detect that the same accessory is in proximity for some period of time, when the owner is not in physical proximity.¶

DULT IETF Draft

3.5.1. Rotation policy

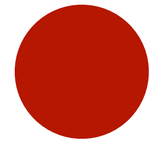
An accessory **SHALL** rotate its address on any transition from near-owner state to separated state as well as any transition from separated state to near-owner state.¶

When in near-owner state, the accessory **SHALL** rotate its address every 15 minutes. This is a privacy consideration to deter tracking of the accessory by non-owners when it is in physical proximity to the owner.¶

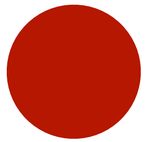
When in a separated state, the accessory **SHALL** rotate its address every 24 hours. This duration allows a platform's unwanted tracking algorithms to detect that the same accessory is in proximity for some period of time, when the owner is not in physical proximity.¶

Is this really all we can do?

Rest of this talk…

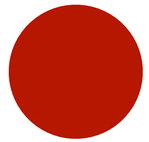


Rest of this talk…



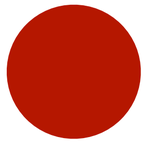
- Describe **one** method for achieving stalker detection with honest user privacy

Rest of this talk…



- Describe **one** method for achieving stalker detection with honest user privacy
  - Will be efficient enough to use in practice*

Rest of this talk…



- Describe **one** method for achieving stalker detection with honest user privacy
  - Will be efficient enough to use in practice*

* According to us

Rest of this talk…

- Describe **one** method for achieving stalker detection with honest user privacy
  - Will be efficient enough to use in practice*

Protocol	Epoch duration	Broadcasts per epoch	Stalker detection?	Tracking privacy	Continuous Proximity	Stalker detection
Apple FindMy [2] / IETF [32]:						
Near-owner mode	15 min	450	✗	n/a	n/a	
Separated mode	24 hrs	43,200	●	n/a	✗	15-60 min†
This work (§4):						
2-second epochs / 1-hour window	2 sec	1	●	40 – 46 min*	●	60 min
4-second epochs / 1-hour window	4 sec	1	●	39 – 46 min*	●	60 min
1-minute epochs / 1-hour window	60 sec	15	●	41 – 47 min*	●	60 min

* According to us

Rest of this talk…

- Describe **one** method for achieving stalker detection with honest user privacy
  - Will be efficient enough to use in practice*

Protocol	Epoch duration	Broadcasts per epoch	Stalker detection?	Tracking privacy	Continuous Proximity	Stalker detection
Apple FindMy [2] / IETF [32]:						
Near-owner mode	15 min	450	✗	n/a	n/a	
Separated mode	24 hrs	43,200	●	n/a	✗	15-60 min†
This work (§4):						
2-second epochs / 1-hour window	2 sec	1	●	40 – 46 min*	●	60 min
4-second epochs / 1-hour window	4 sec	1	●	39 – 46 min*	●	60 min
1-minute epochs / 1-hour window	60 sec	15	●	41 – 47 min*	●	60 min

* According to us

What makes this problem hard?

What makes this problem hard?

- Potential stalking victim and tracking adversary have the same goal

What makes this problem hard?

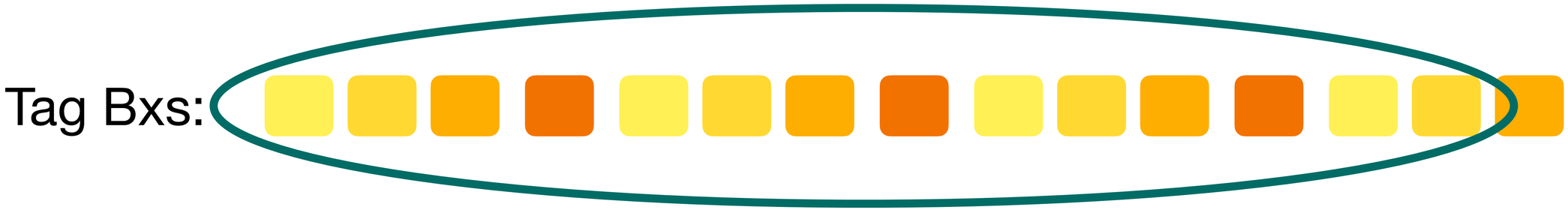
- Potential stalking victim and tracking adversary have the same goal
  - want to detect **repeated contact** with the **same** tag

Tracking adv. and stalking victims are not the same

Tag Bxs:

Tracking adv. and stalking victims are not the same

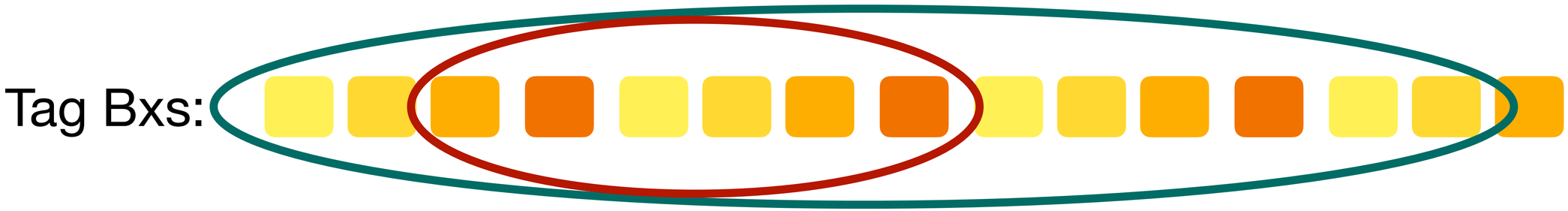
Seen by stalking victim



Tracking adv. and stalking victims are not the same

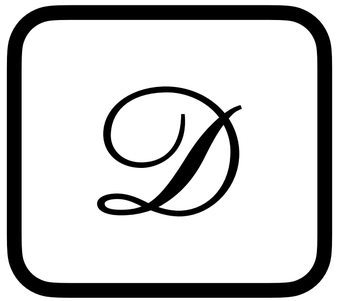
Seen by stalking victim

Seen by tracking adversary

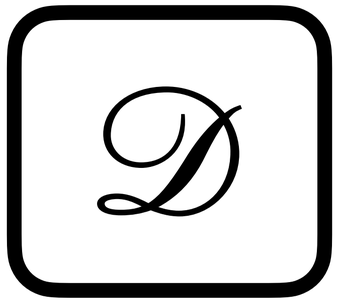


A new primitive: Multi-Dealer Secret Sharing

A new primitive: Multi-Dealer Secret Sharing

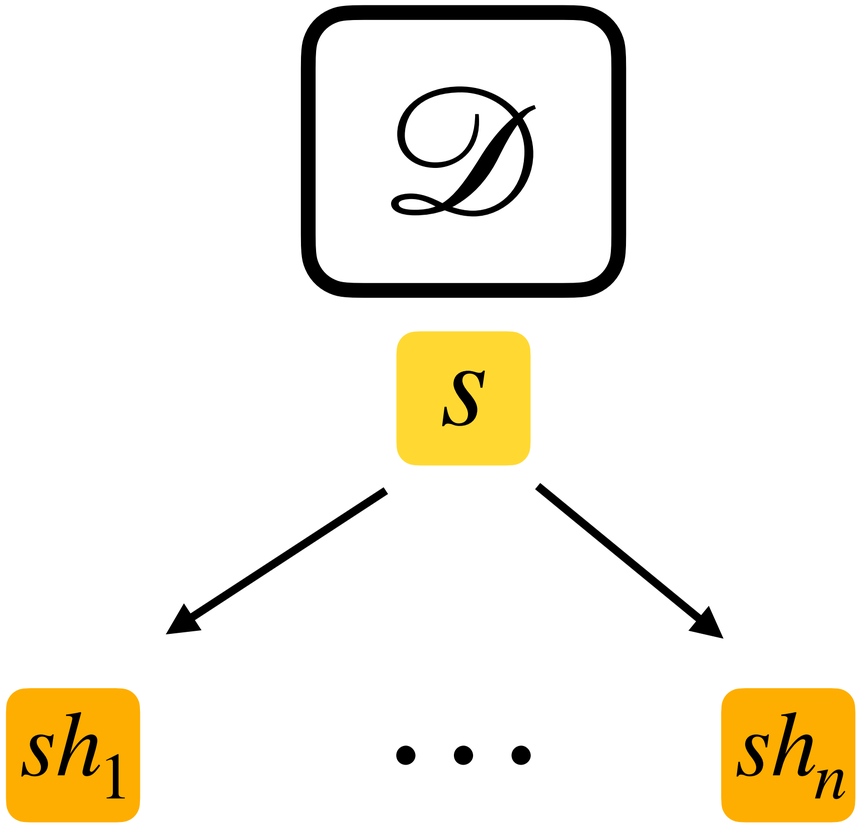


A new primitive: Multi-Dealer Secret Sharing



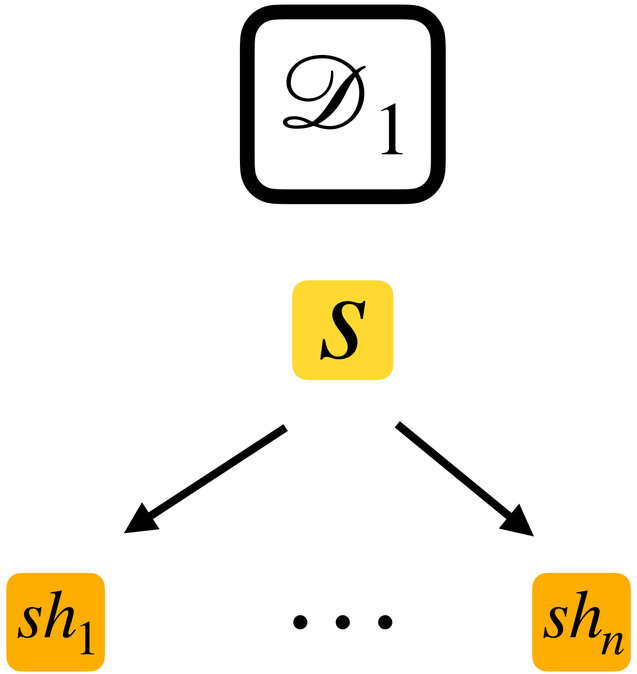
S

A new primitive: Multi-Dealer Secret Sharing

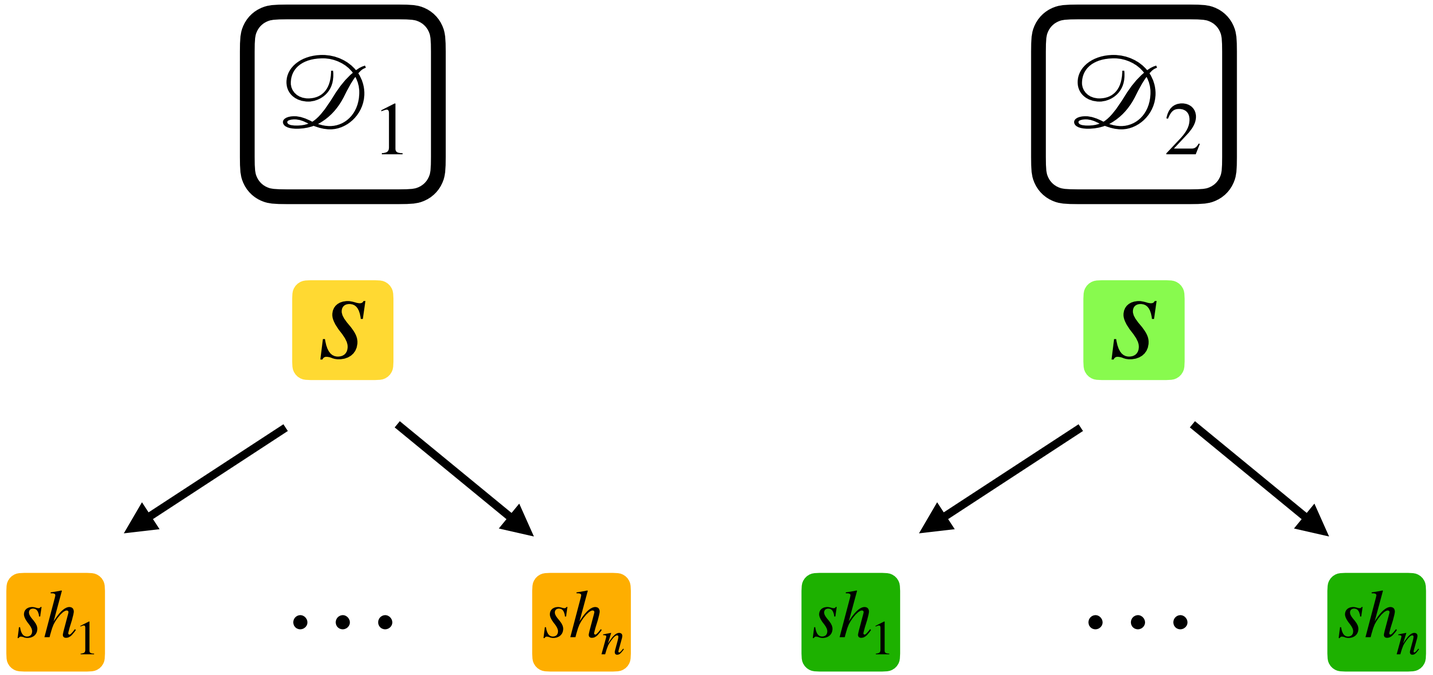


A new primitive: Multi-Dealer Secret Sharing

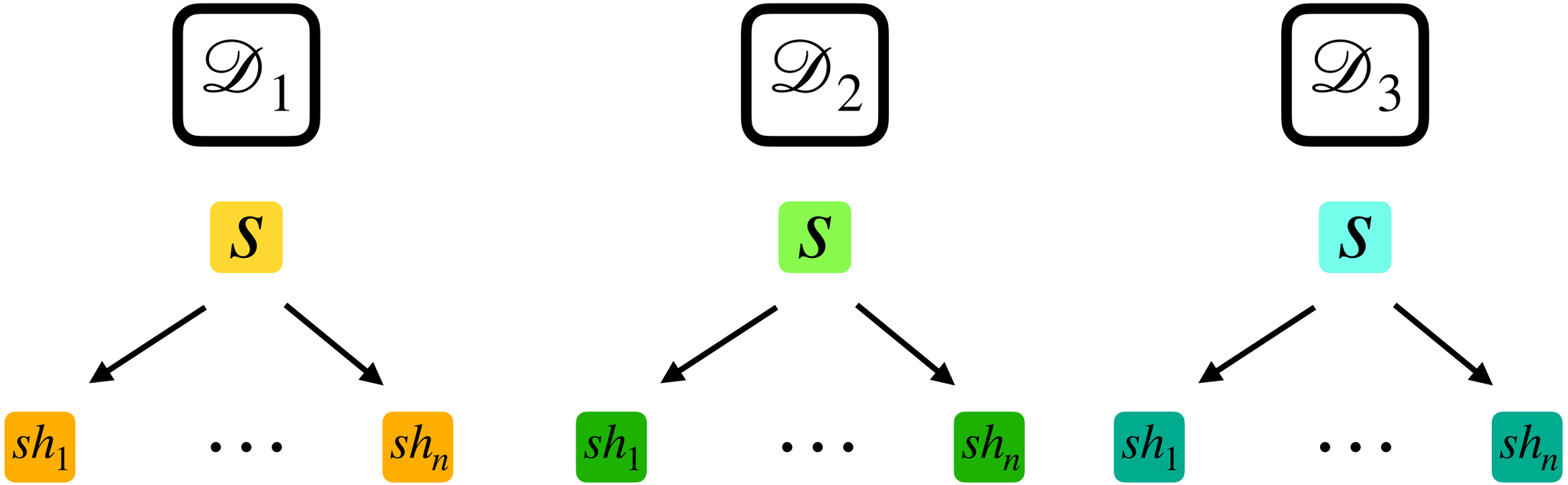
A new primitive: Multi-Dealer Secret Sharing



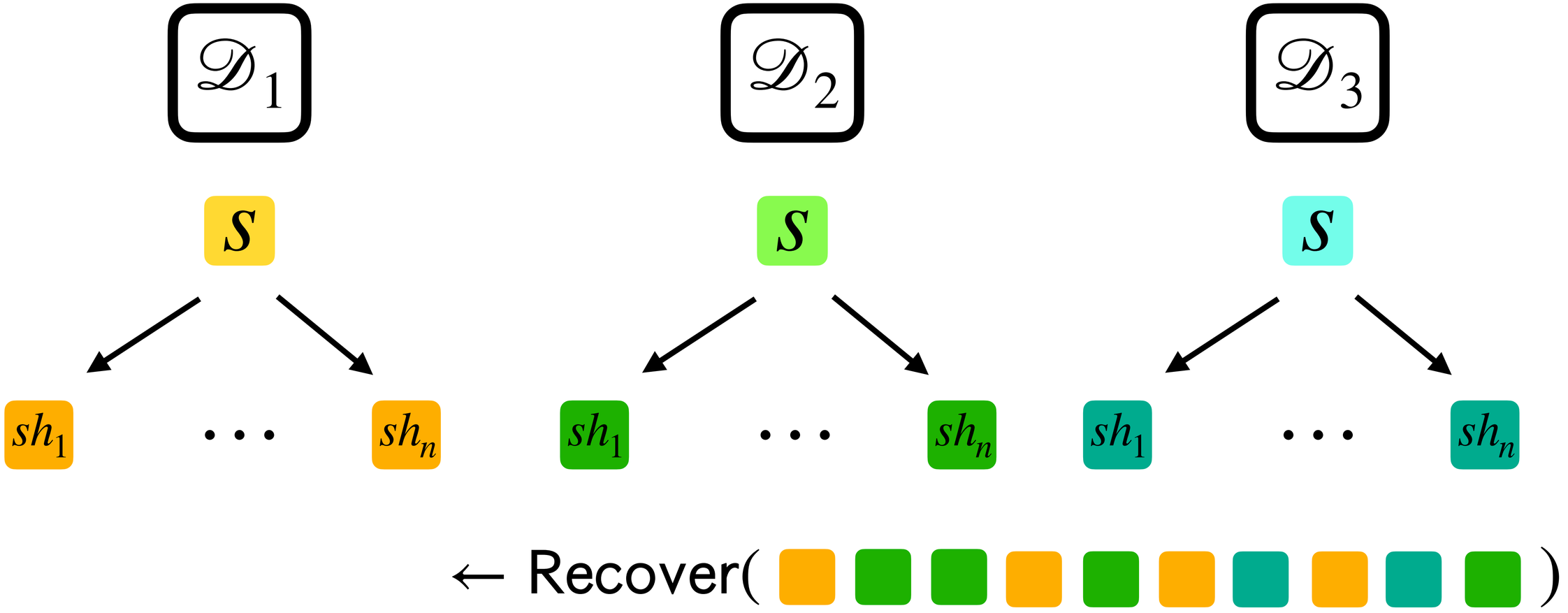
A new primitive: Multi-Dealer Secret Sharing



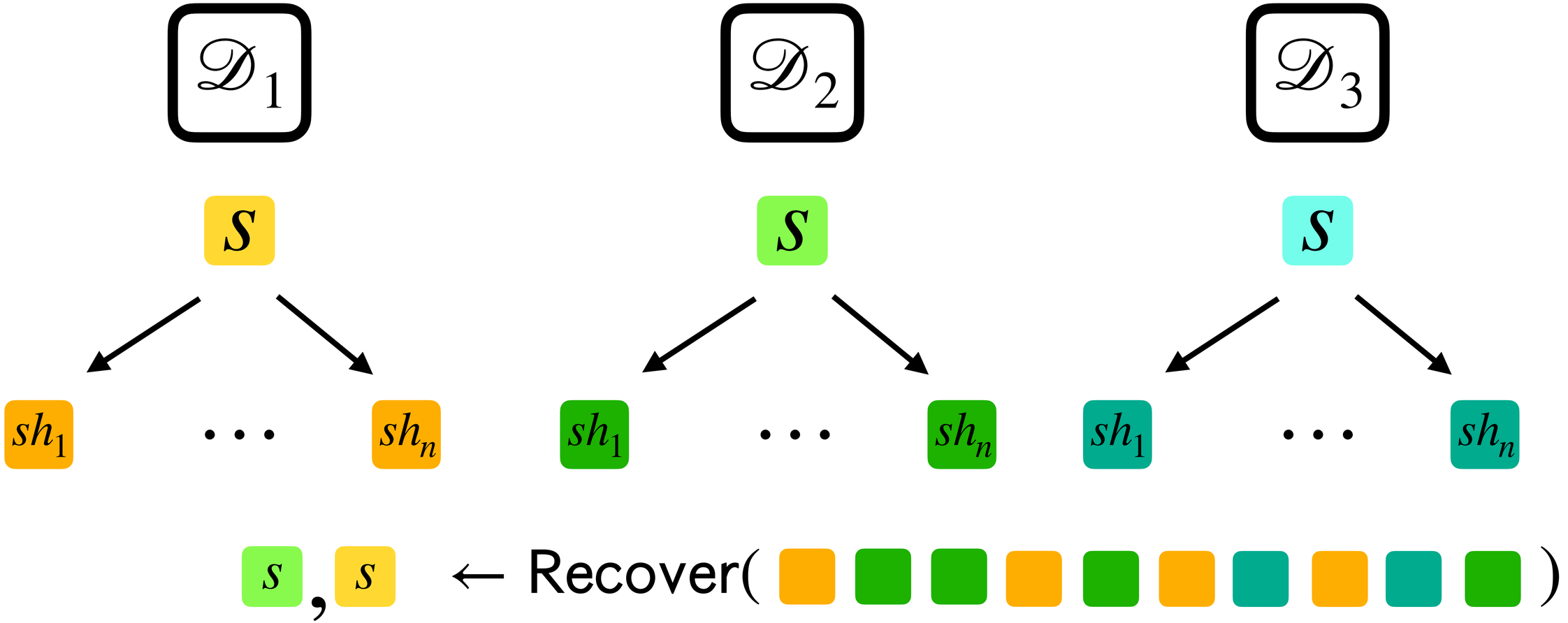
A new primitive: Multi-Dealer Secret Sharing



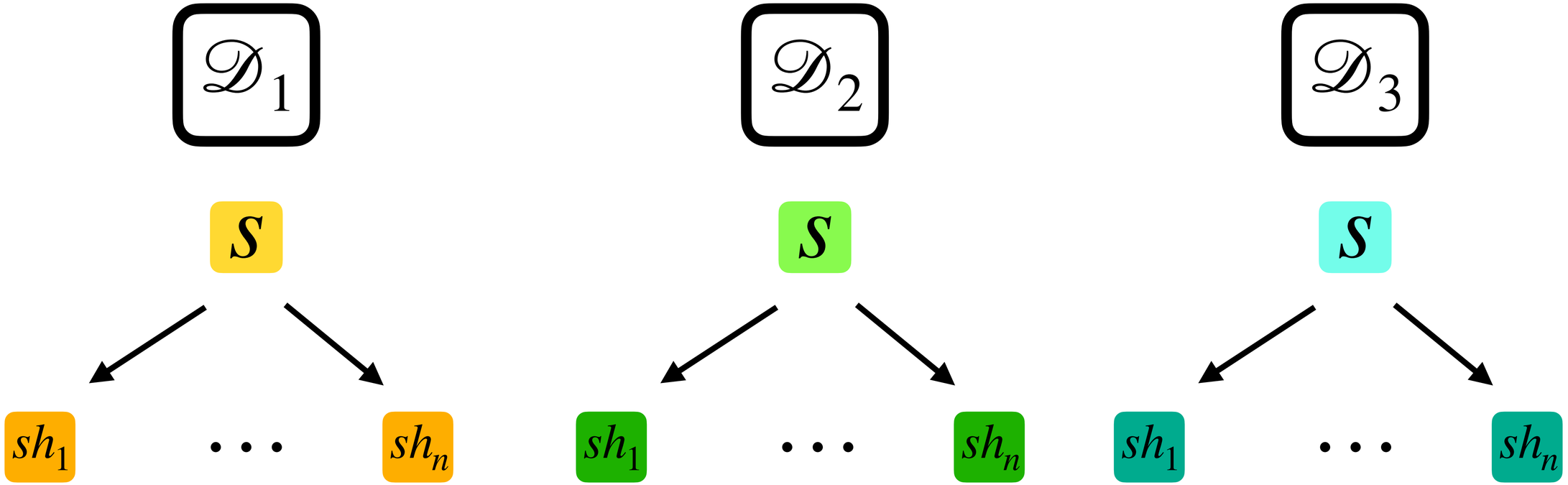
A new primitive: Multi-Dealer Secret Sharing



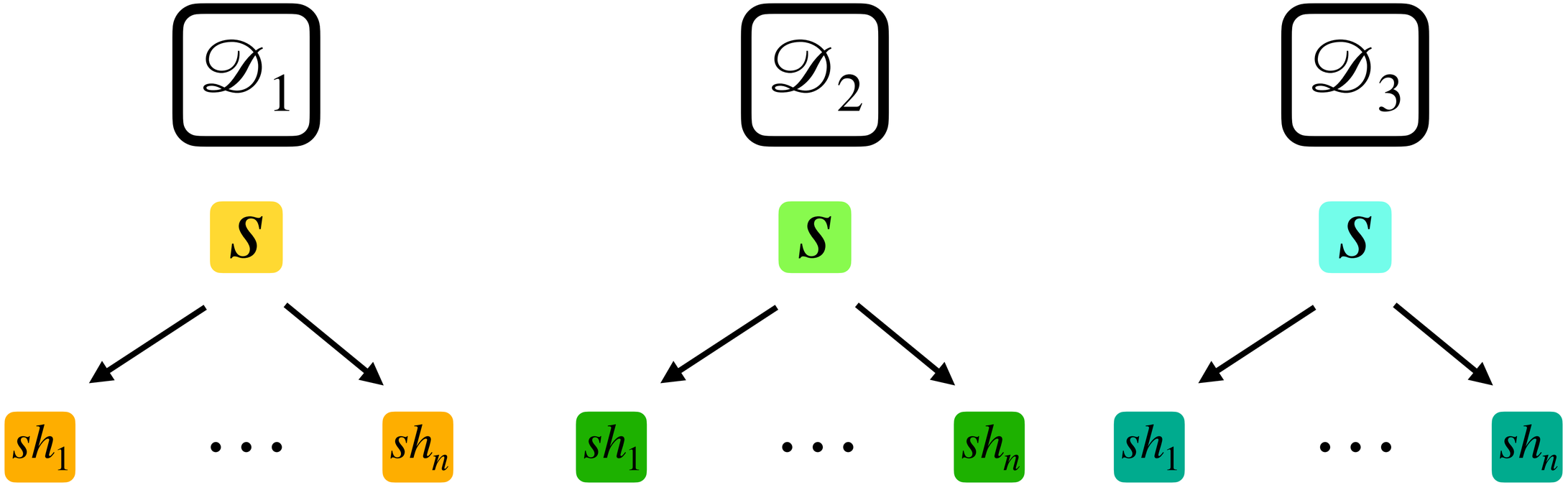
A new primitive: Multi-Dealer Secret Sharing



A new primitive: Multi-Dealer Secret Sharing

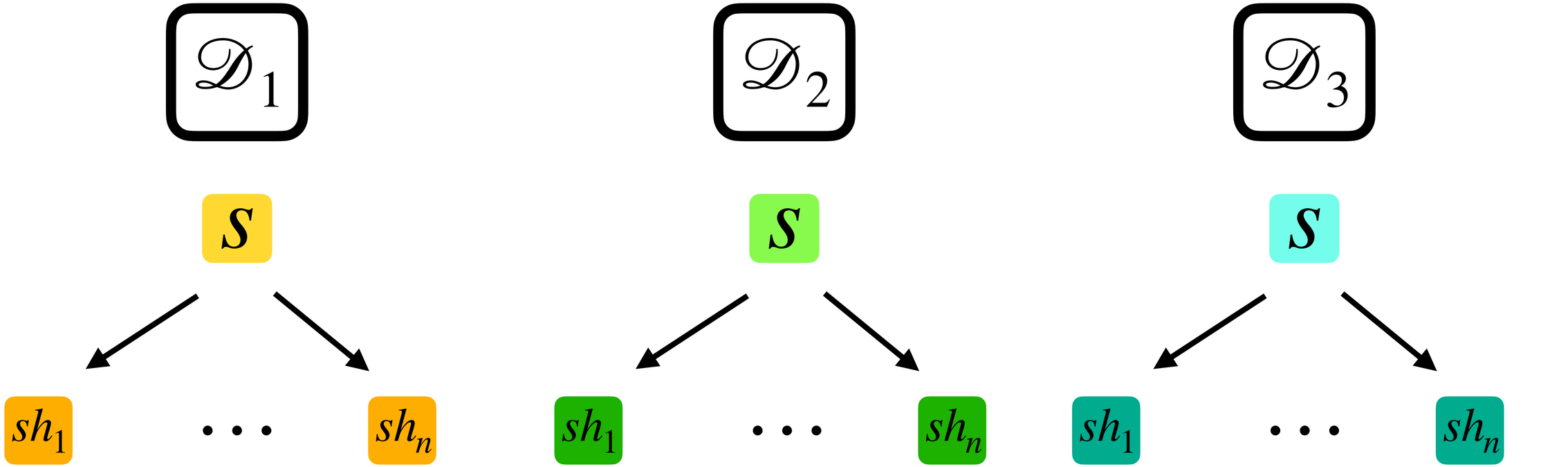


A new primitive: Multi-Dealer Secret Sharing



New property, *unlinkability:*

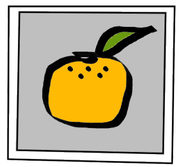
A new primitive: Multi-Dealer Secret Sharing



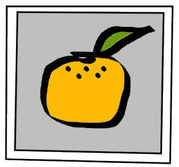
New property, *unlinkability:*

How can MDSS help?

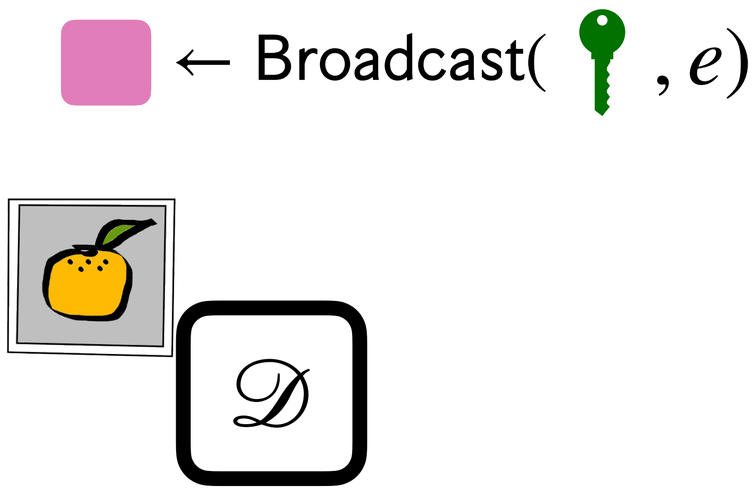
How can MDSS help?



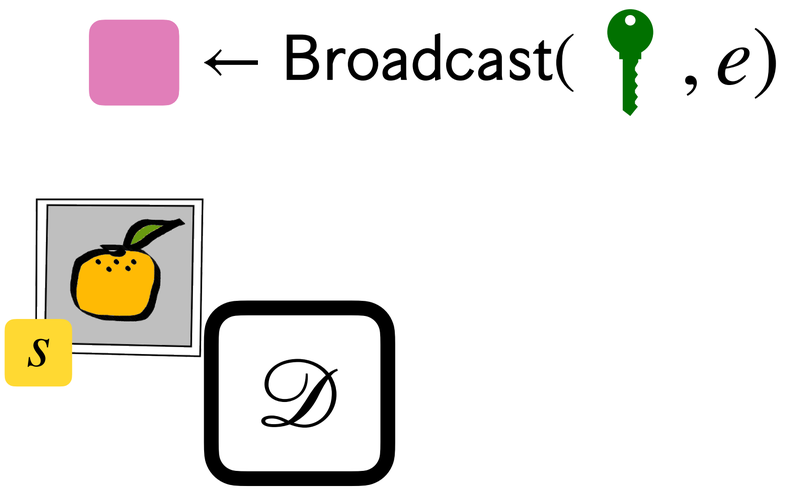
How can MDSS help?



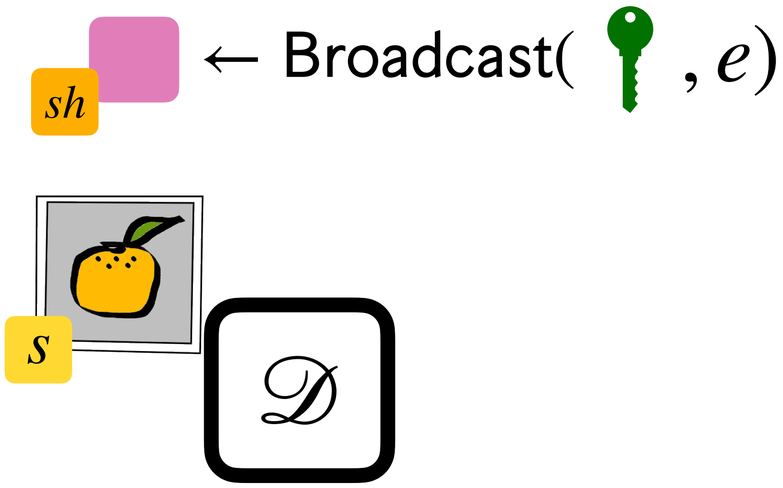
How can MDSS help?



How can MDSS help?

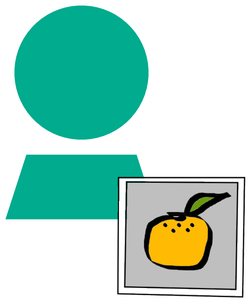


How can MDSS help?

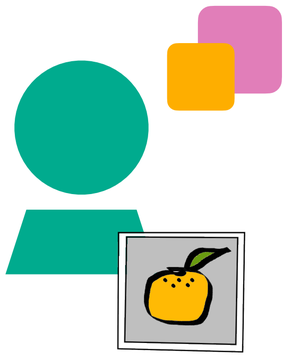


How can MDSS help?

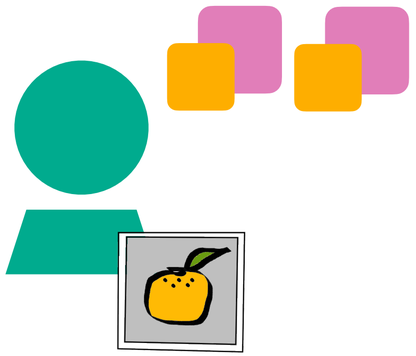
How can MDSS help?



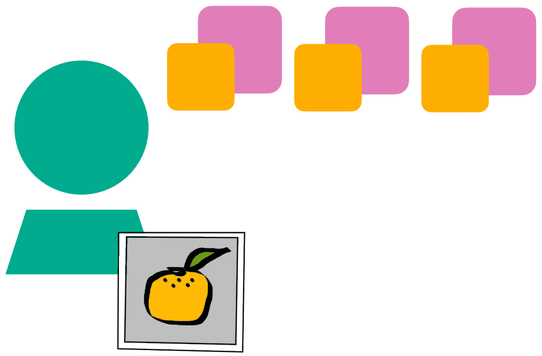
How can MDSS help?



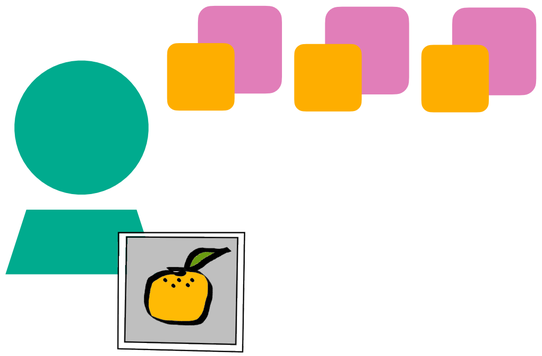
How can MDSS help?

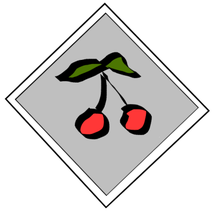


How can MDSS help?

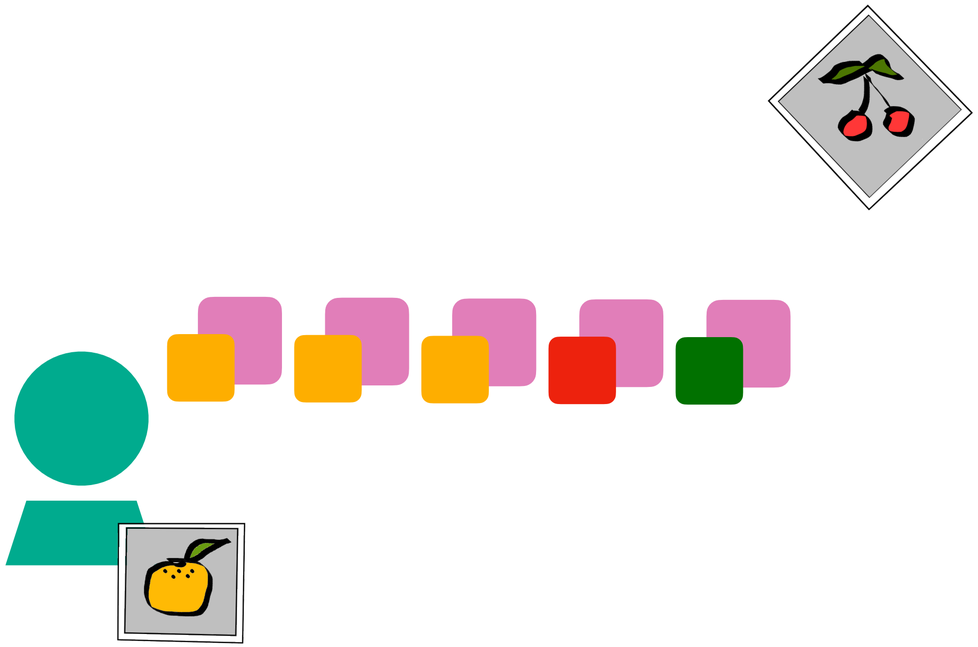


How can MDSS help?

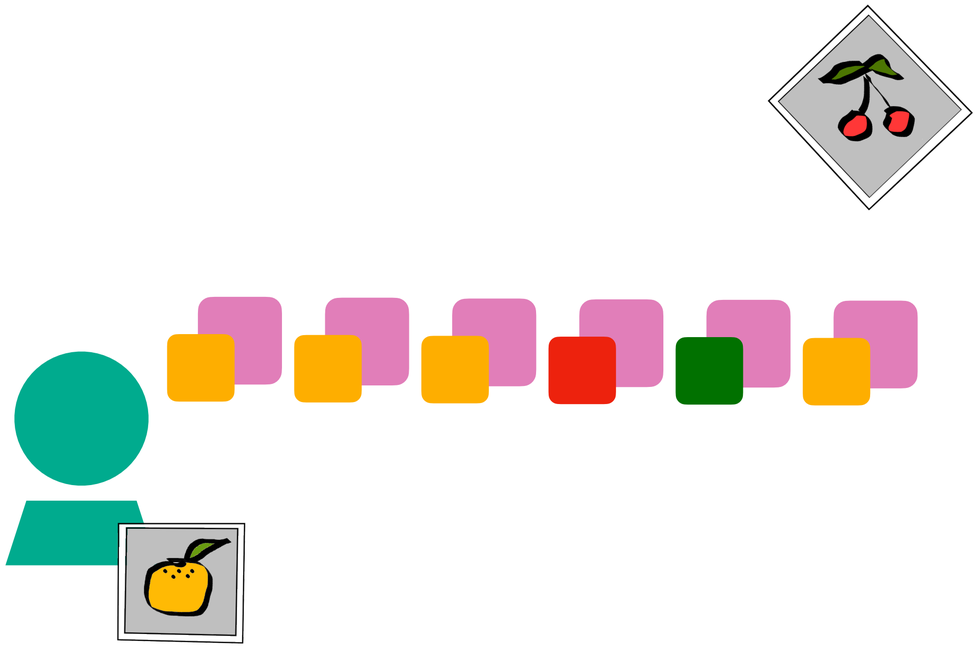




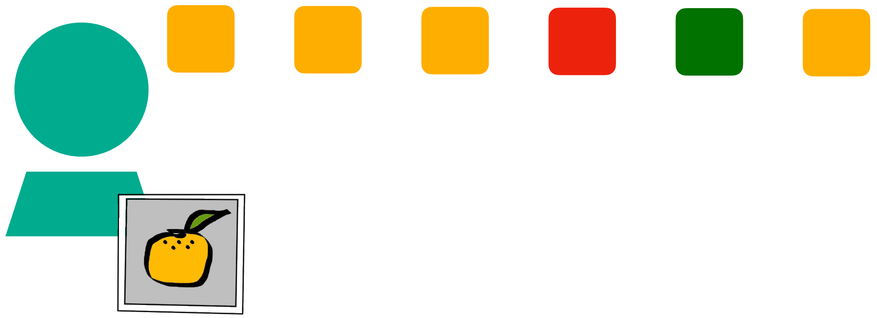
How can MDSS help?



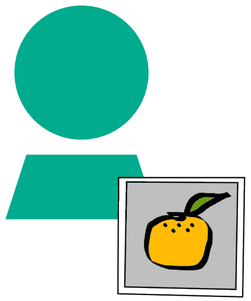
How can MDSS help?



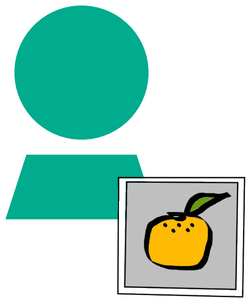
How can MDSS help?



How can MDSS help?

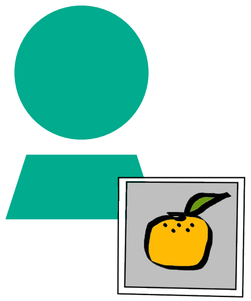


How can MDSS help?



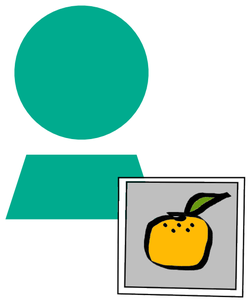
MDSS . Recover()

How can MDSS help?



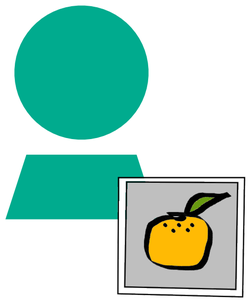
MDSS . Recover() →

How can MDSS help?

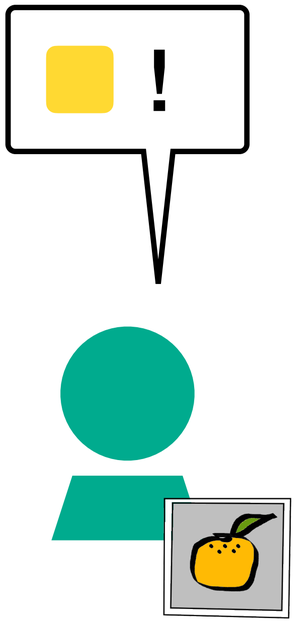


MDSS . Recover()

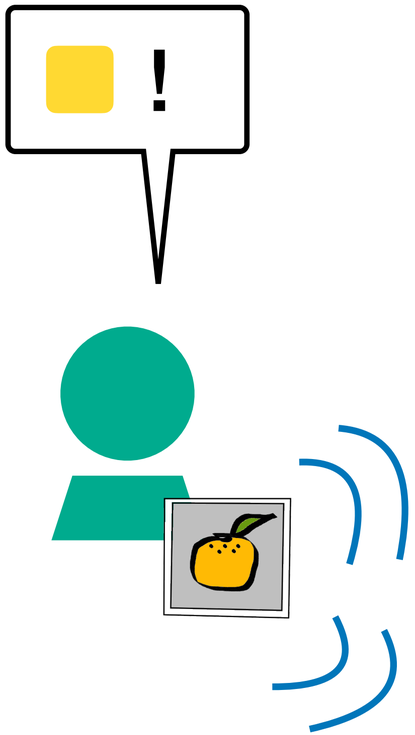
How can MDSS help?



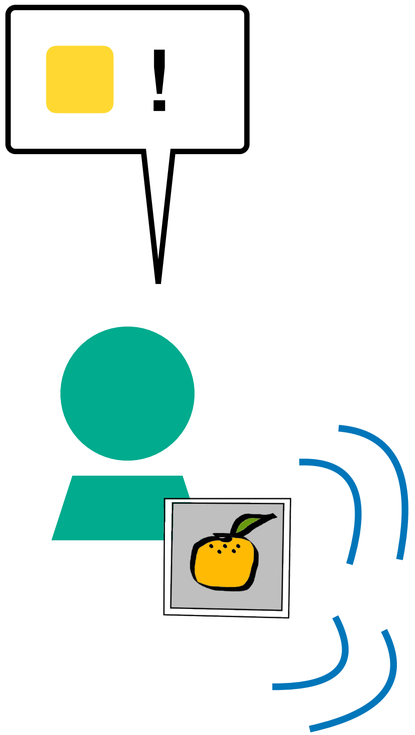
How can MDSS help?



How can MDSS help?



How can MDSS help?



Privacy for honest users follows from MDSS unlinkability property

How to construct MDSS?

How to construct MDSS?

- Easy to construct **a** MDSS scheme…

How to construct MDSS?

- Easy to construct **a** MDSS scheme…
  - Use Shamir secret-sharing where $sh_i = (\alpha_i, p(\alpha_i))$

How to construct MDSS?

- Easy to construct **a** MDSS scheme…
  - Use Shamir secret-sharing where $sh_i = (\alpha_i, p(\alpha_i))$
    - unlinkable when α_i sampled **uniformly** at random

How to construct MDSS?

- Easy to construct **a** MDSS scheme…
  - Use Shamir secret-sharing where $sh_i = (\alpha_i, p(\alpha_i))$
    - unlinkable when α_i sampled **uniformly** at random
    - do recovery using RS **list-decoding** algorithm **[GS98]**

How to construct MDSS?

- Easy to construct **a** MDSS scheme…
  - Use Shamir secret-sharing where $sh_i = (\alpha_i, p(\alpha_i))$
    - unlinkable when α_i sampled **uniformly** at random
    - do recovery using RS **list-decoding** algorithm **[GS98]**
      - ** works if dealer's shares make up $\geq (1 - \rho)$ fraction of input

Unfortunately…

Unfortunately…

- Unlinkability for MDSS degrades quickly with more dealers

Unfortunately…

- Unlinkability for MDSS degrades quickly with more dealers
- To avoid this …

Unfortunately…

- Unlinkability for MDSS degrades quickly with more dealers
- To avoid this …
  - Need to list decode near capacity

Unfortunately…

- Unlinkability for MDSS degrades quickly with more dealers
- To avoid this …
  - Need to **list decode** near **capacity**

A better MDSS scheme

A better MDSS scheme

- What if we use a different type of RS code?

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…
    - Proofs only for **random** error model

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…
    - Proofs only for **random** error model 😶

A better MDSS scheme

- What if we use a different type of RS code?
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…
    - Proofs only for **random** error model 😶

A better MDSS scheme

Modify **[CH11]** to (*heuristically*) handle recovering multiple dealers secrets

- What if we use a d
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…
    - Proofs only for **random** error model 😶

A better MDSS scheme

Modify **[CH11]** to (*heuristically*) handle recovering multiple dealers secrets

(If you're interested in this come talk to me!! Please!)

- What if we use a d
  - $sh_i = (\alpha, p_1(\alpha), \ldots p_c(\alpha))$
- **[BKY03, CS03, CH11]** - by increasing c, **could** avoid unlinkability degradation
  - But…
    - Proofs only for **random** error model 😮

Efficiency

- Most aggressive parameters of **4 sec** epoch duration
  - Stalker detection on laptop < 30 sec (No stalkers < 4 sec), Raspberry Pi < 6 min (No stalkers < 1 min)
- For **60 sec** epoch duration
  - Raspberry Pi < 1.27 sec (No stalkers .48 sec)
- Reminder: Apple's epoch duration is **15 min**

Efficiency

- Most aggressive parameters of **4 sec** epoch duration
  - Stalker detection on laptop < 30 sec (No stalkers < 4 sec), Raspberry Pi < 6 min (No stalkers < 1 min)
- For **60 sec** epoch duration
  - Raspberry Pi < 1.27 sec (No stalkers .48 sec)
- Reminder: Apple's epoch duration is **15 min**

In conclusion…

- Not impossible to come up with schemes which allow for stalker detection while still giving privacy to honest tag users
- Would be nice to see other solutions in this space

Eprint: https://eprint.iacr.org/2023/1332

Dying social media platform: https://twitter.com/gabrie_beck