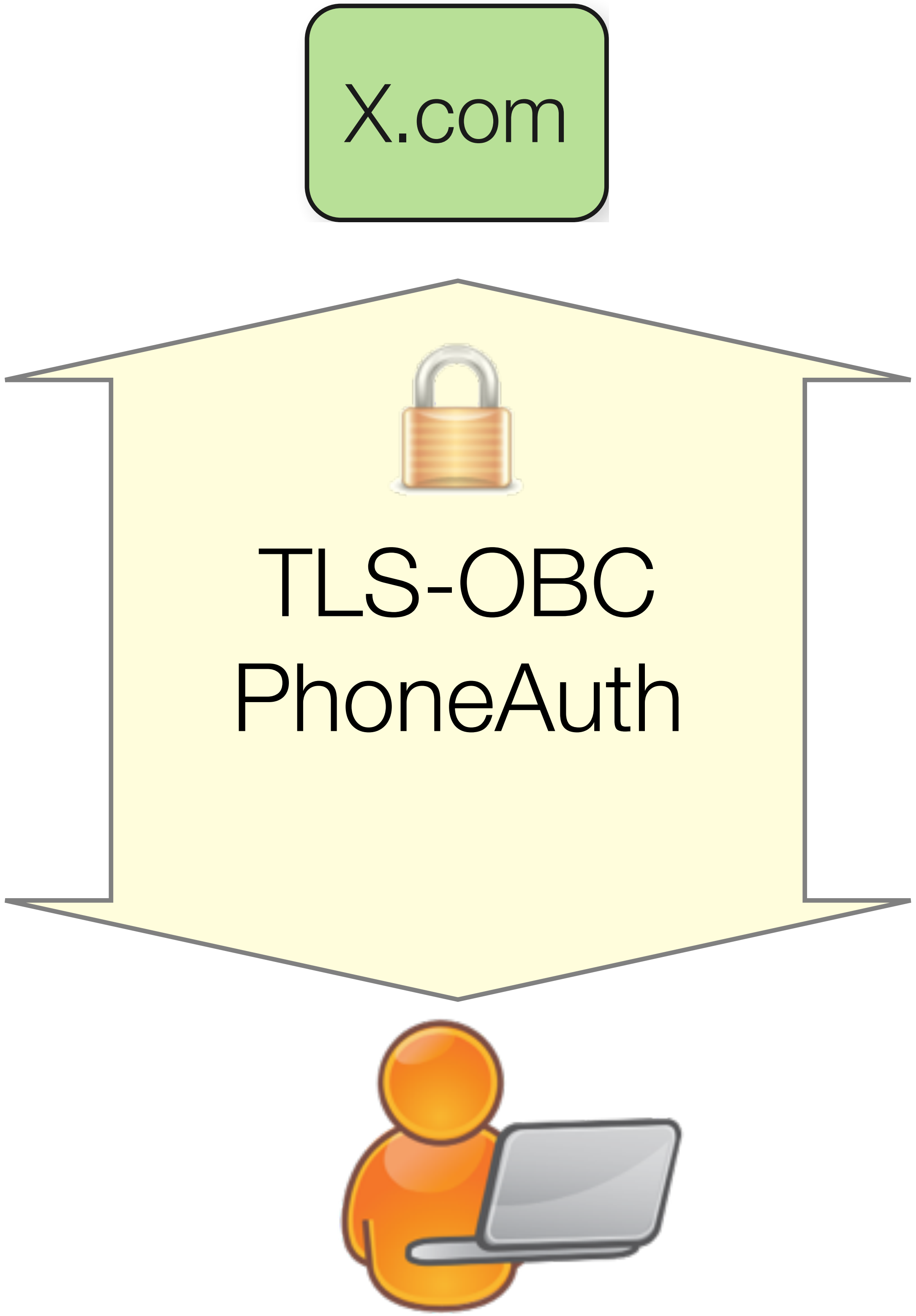
Hardening Persona: Improving Federated Login on the Web

Michael Dietz
and
Dan S. Wallach

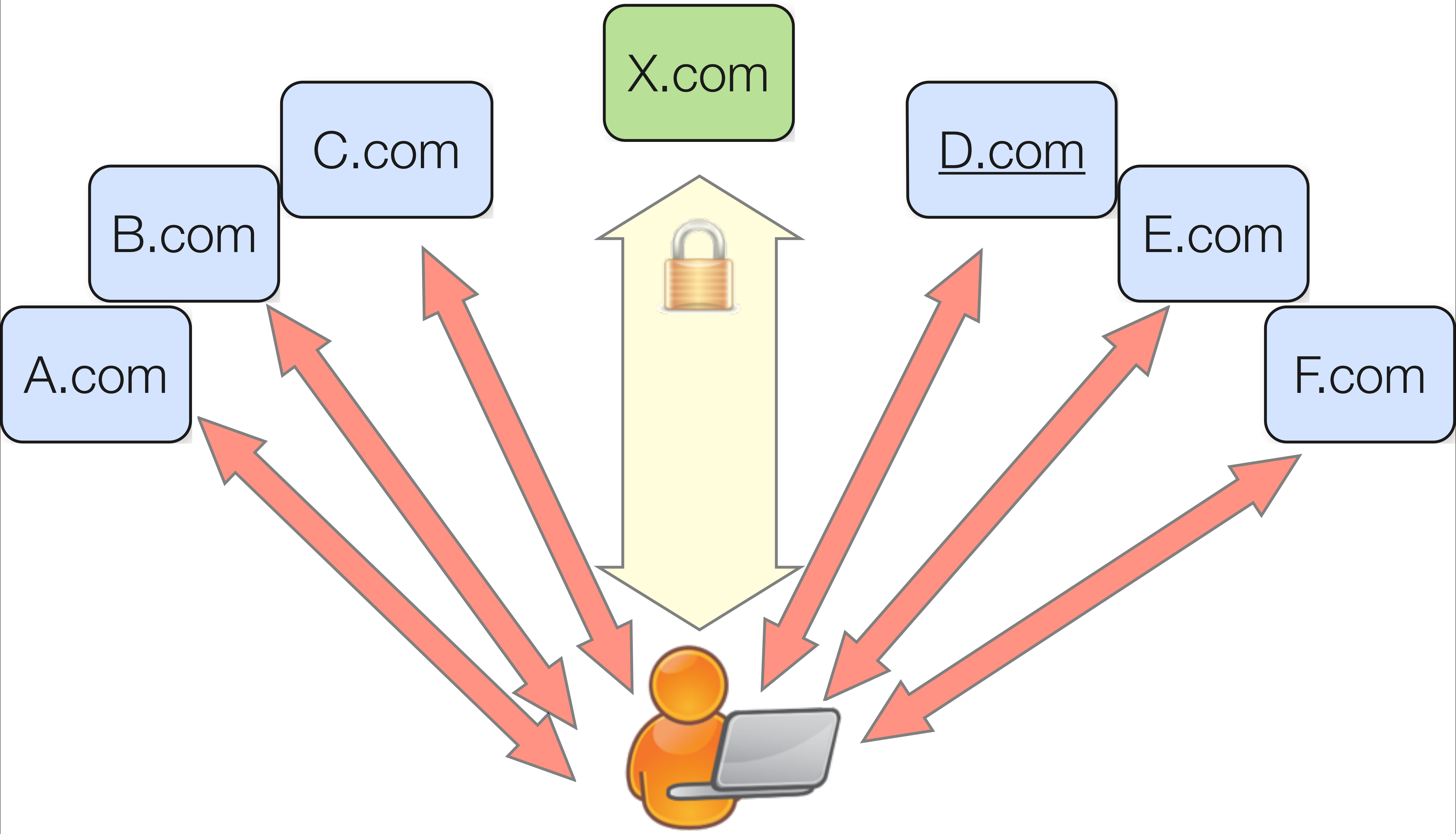
RICE
Unconventional Wisdom

COMPUTER SECURITY LAB
Rice University

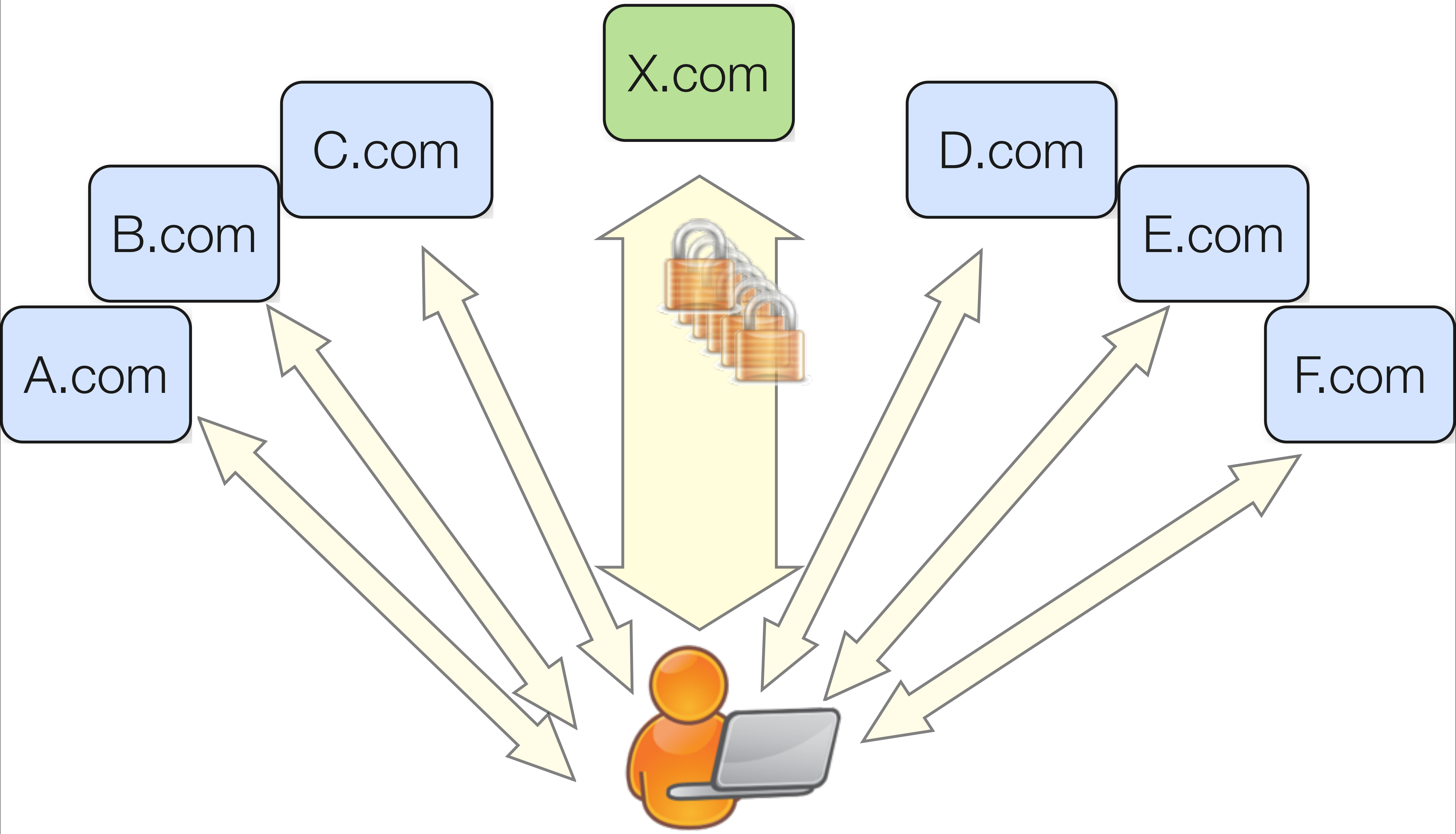
Motivation



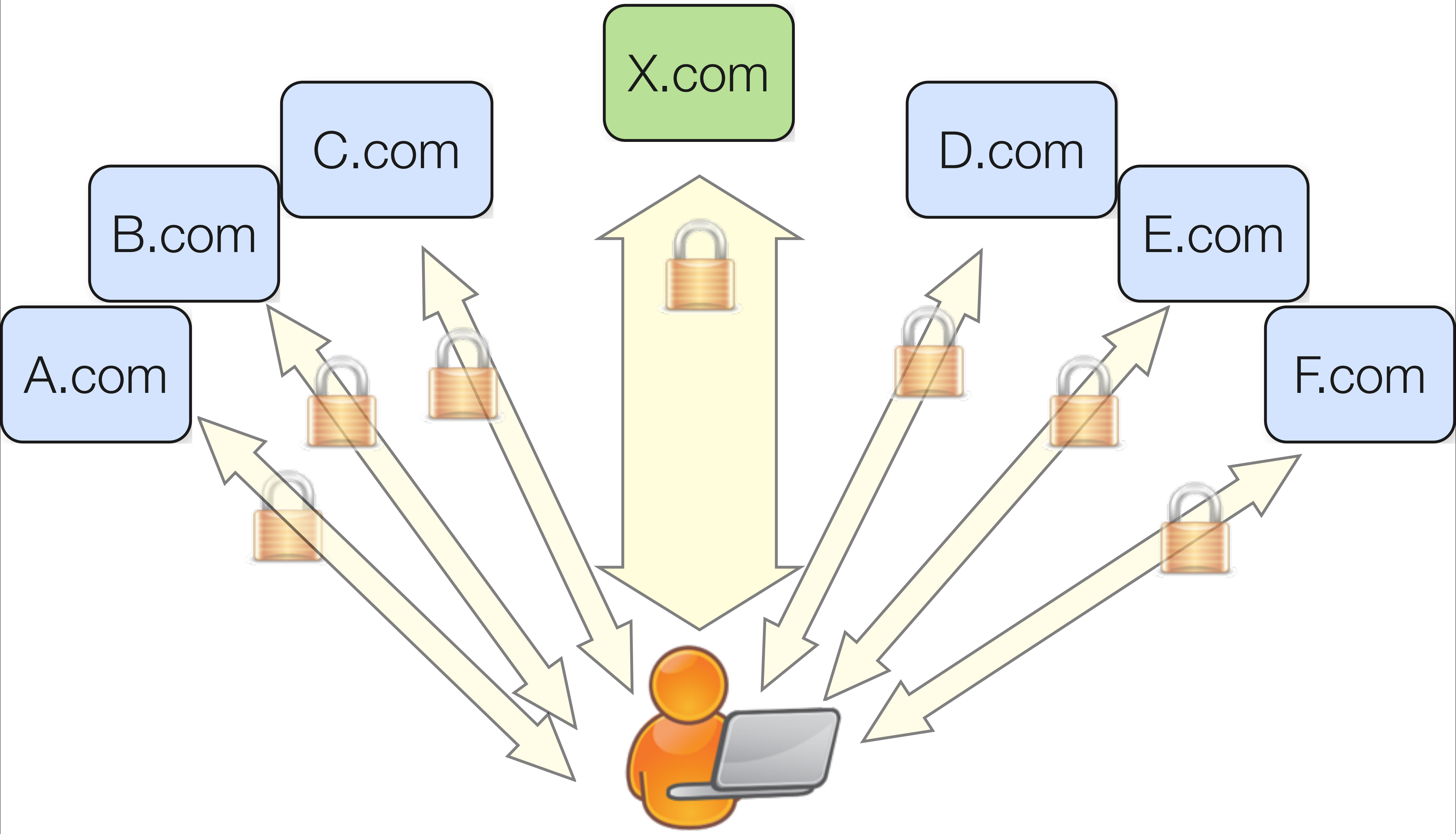
Motivation



Motivation



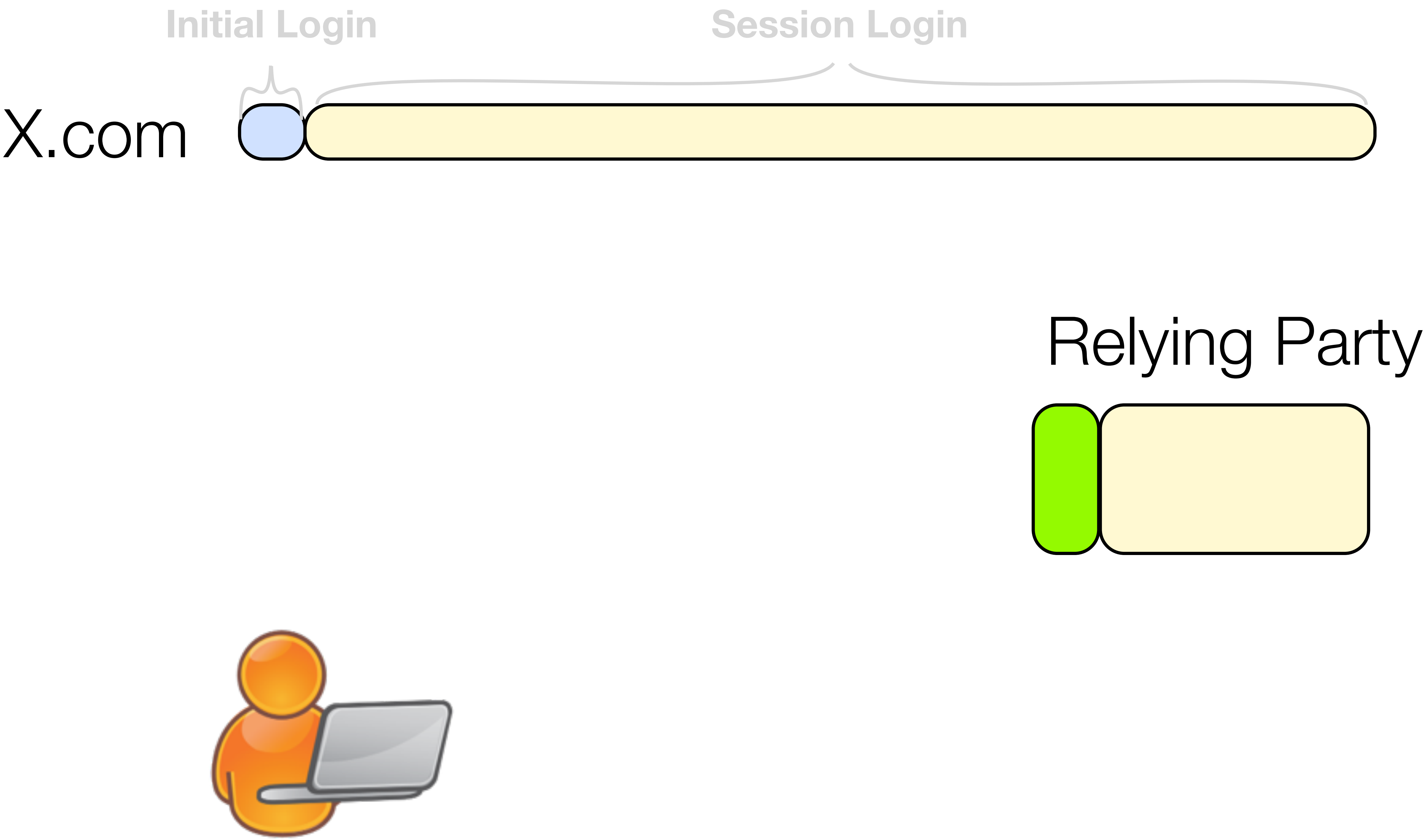
Motivation



Existing federated login protocols

- SAML
- OpenID
- Persona (BrowserId)
- OAuth
- OAuth2 / OpenIdConnect
- Kerberos

Persona Overview



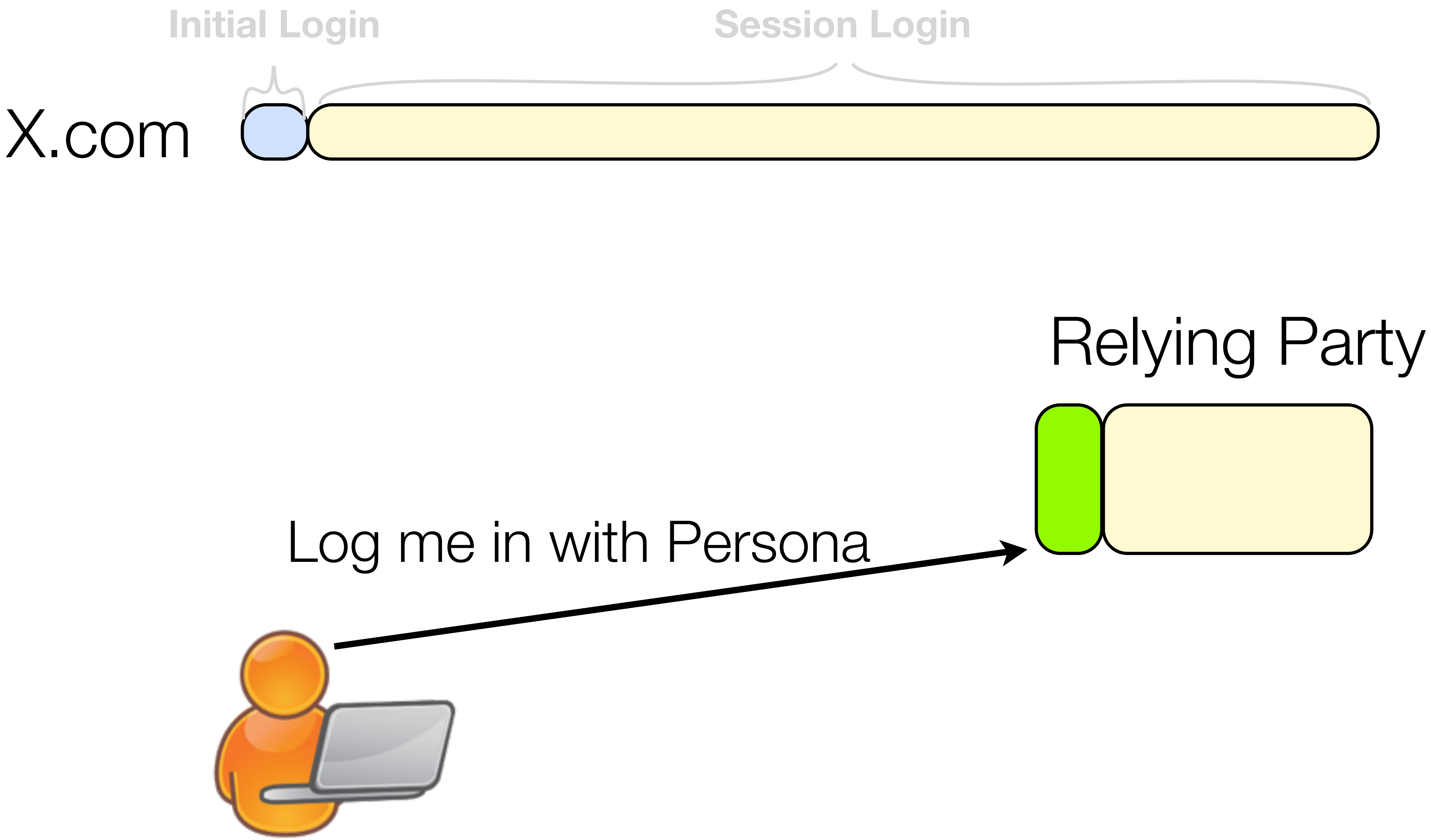
Initial Login

Session Login

X.com

Relying Party

Persona Overview



Initial Login

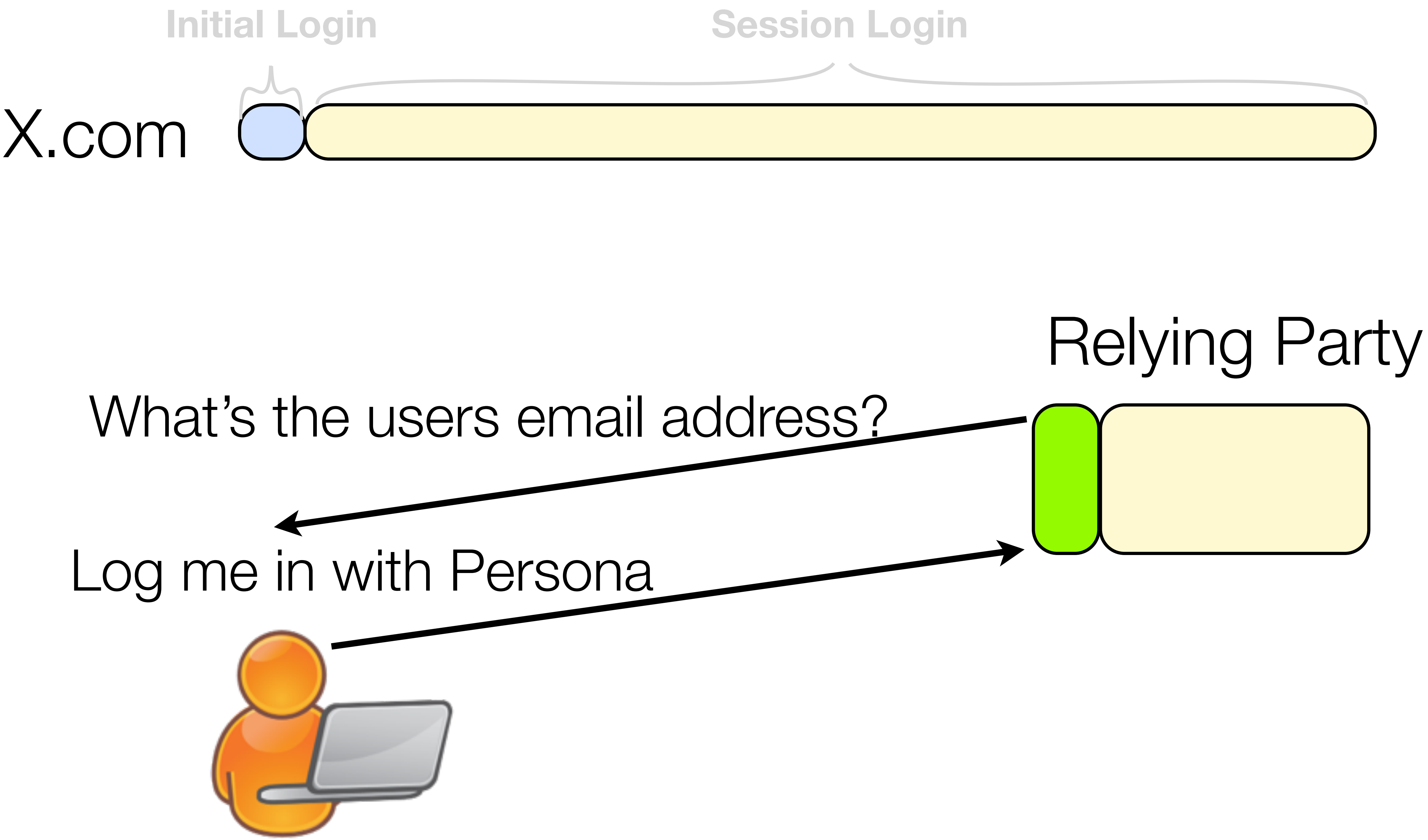
Session Login

X.com

Relying Party

Log me in with Persona

Persona Overview



Initial Login

Session Login

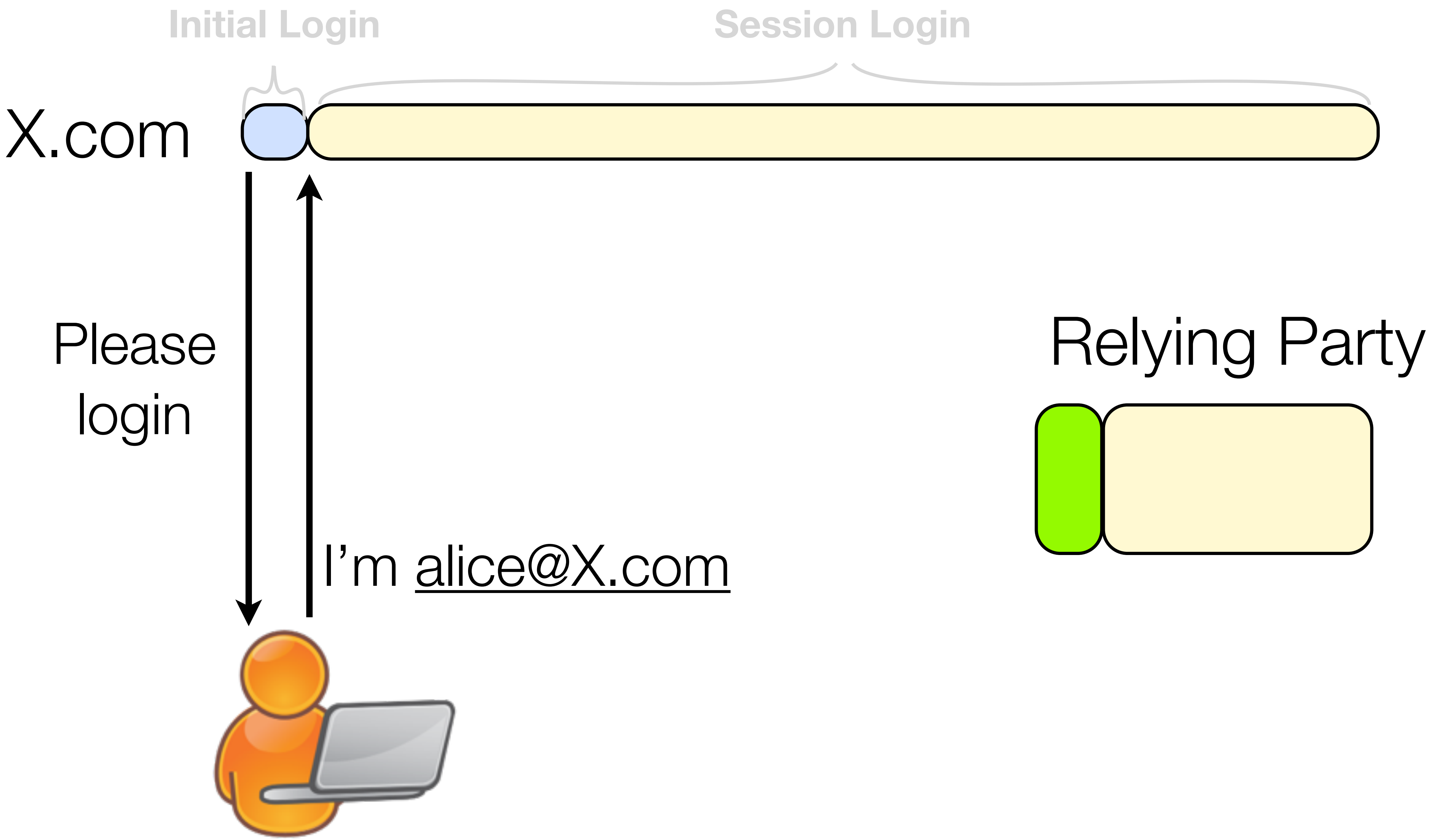
X.com

Relying Party

What’s the users email address?

Log me in with Persona

Persona Overview



Initial Login

Session Login

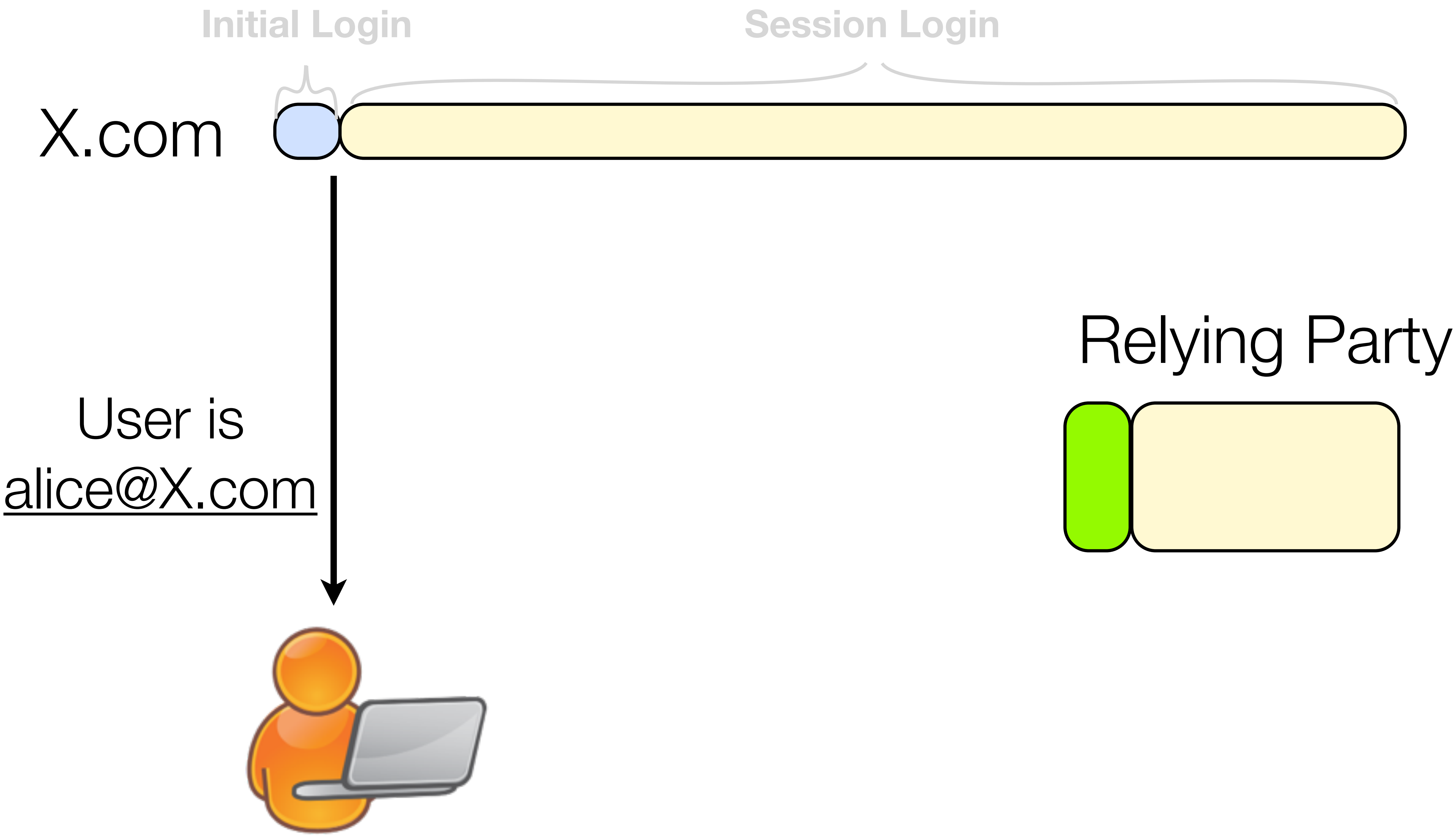
X.com

Please login

I’m alice@X.com

Relying Party

Persona Overview



Initial Login

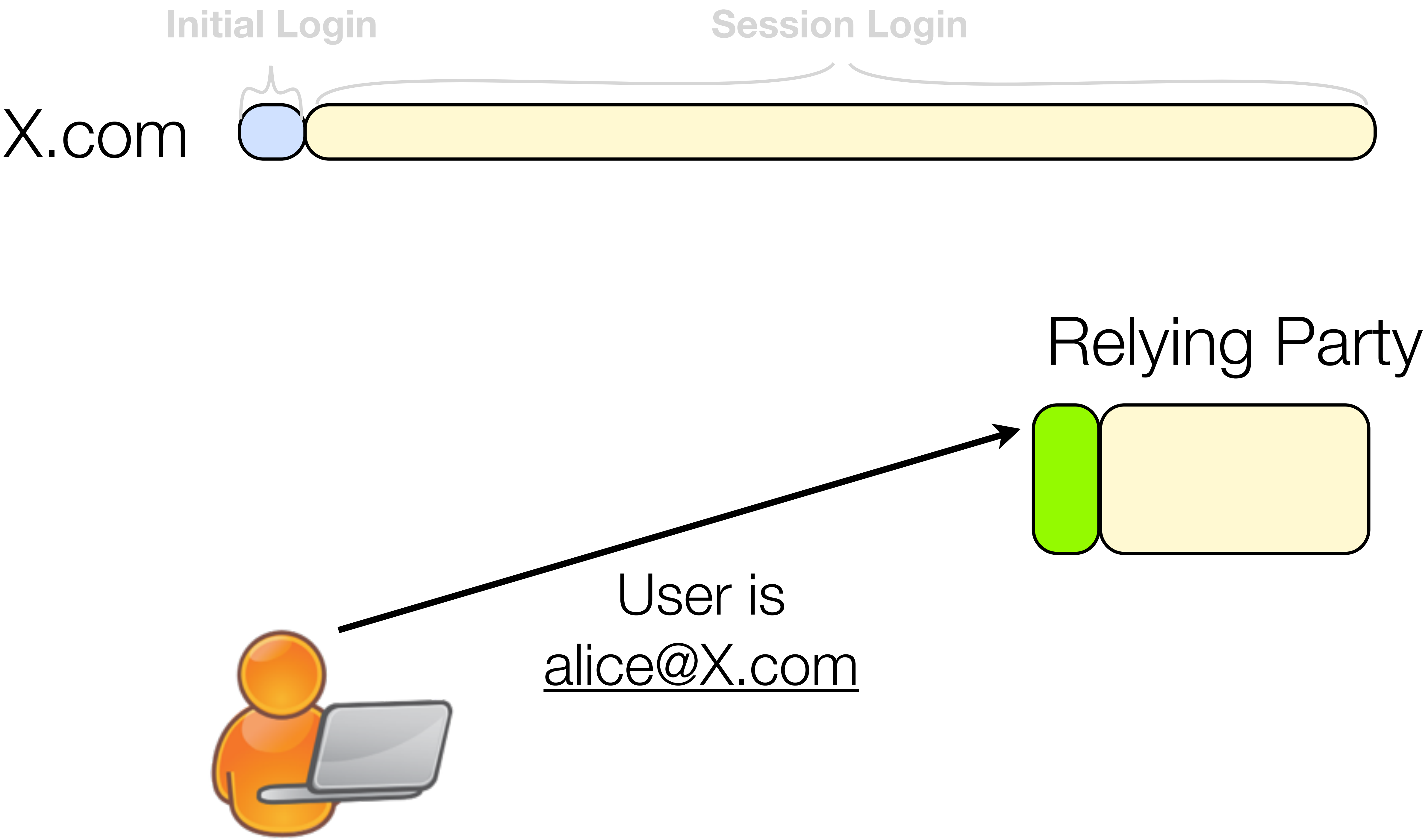
Session Login

X.com

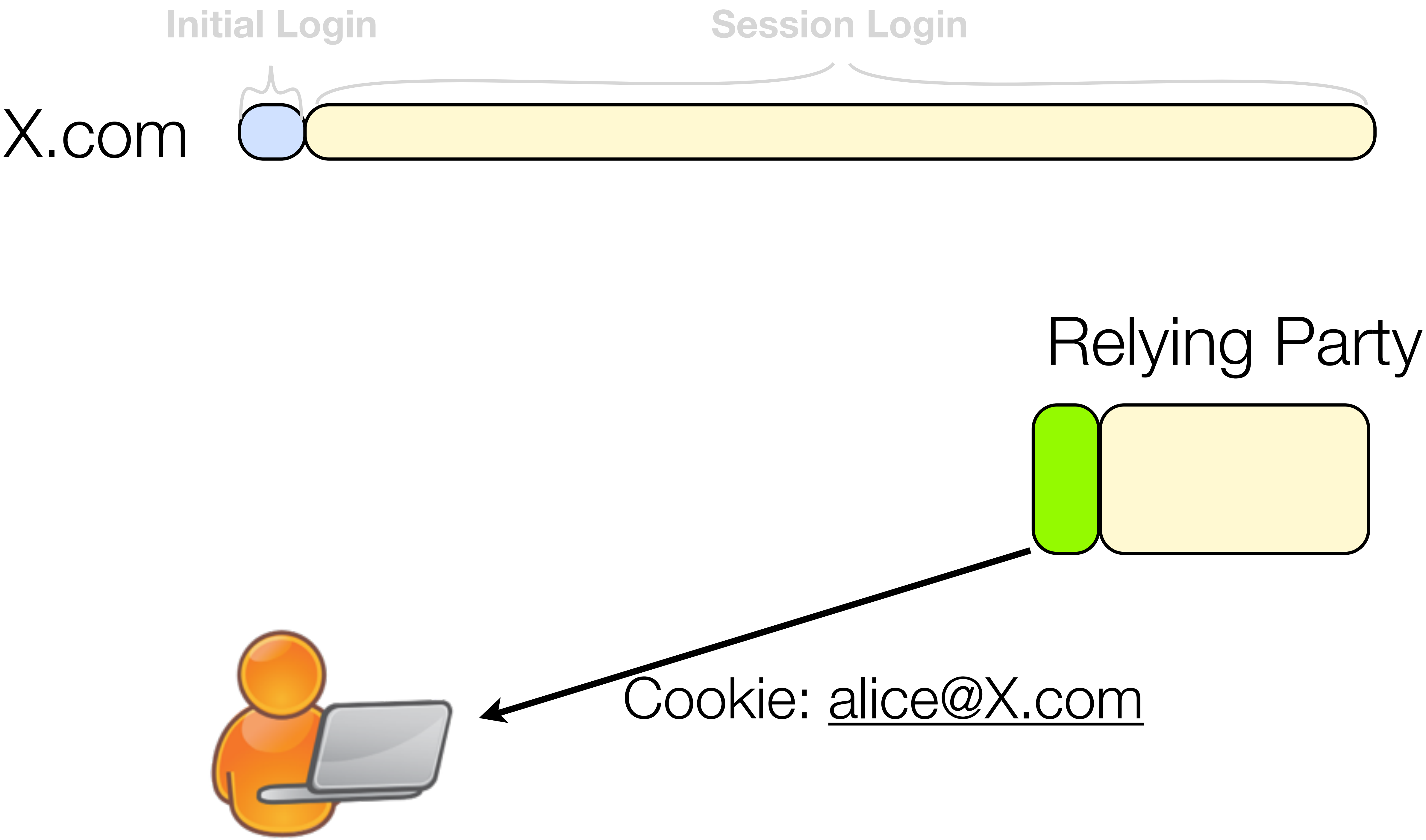
User is alice@X.com

Relying Party

Persona Overview



Persona Overview



Initial Login

Session Login

X.com

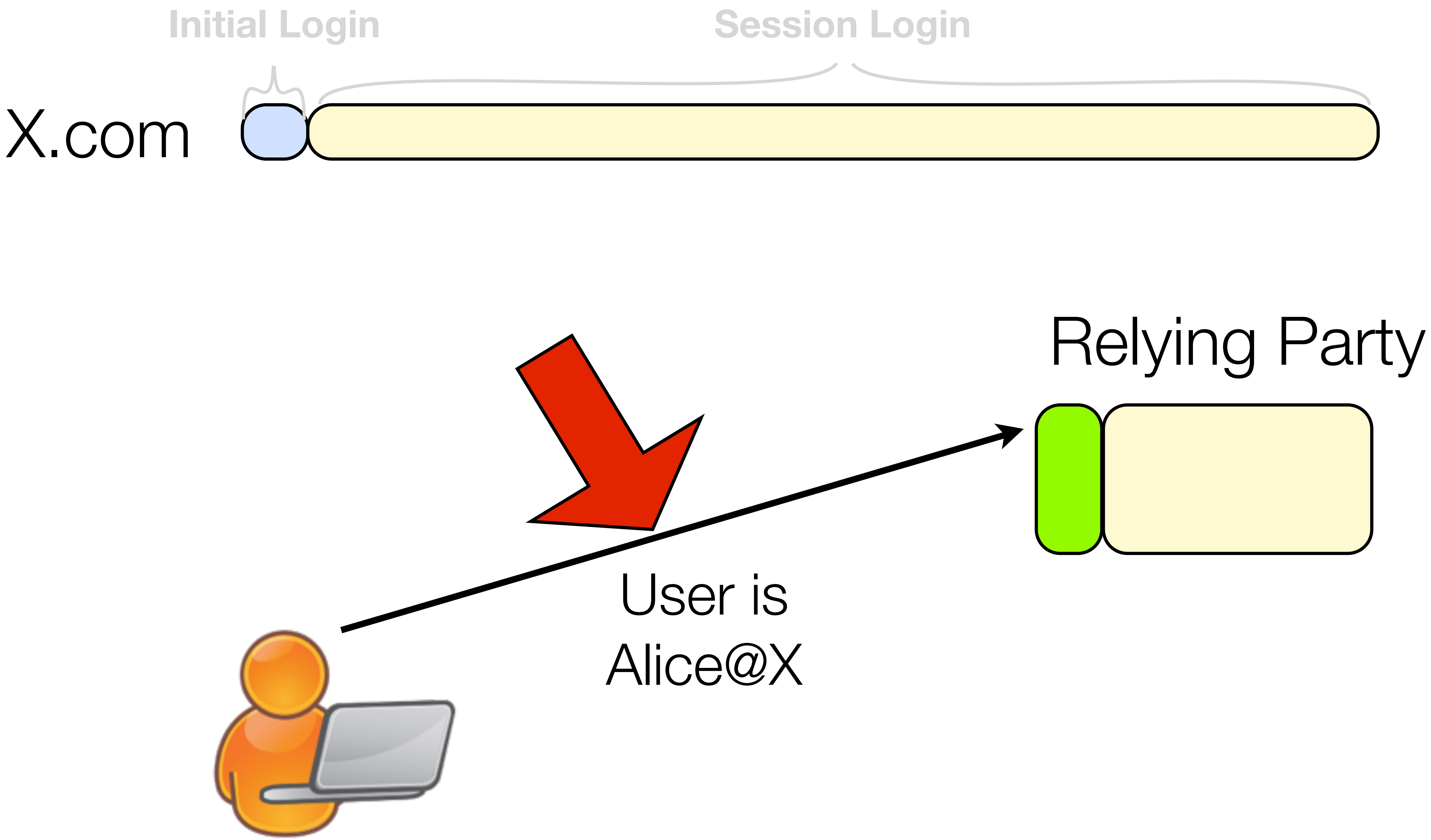
Relying Party

Cookie: alice@X.com

Two areas for attack

- MITM the connection between user and RP
    - Replay identity assertions
- Steal relying party cookie after login

Identity assertion theft



Initial Login

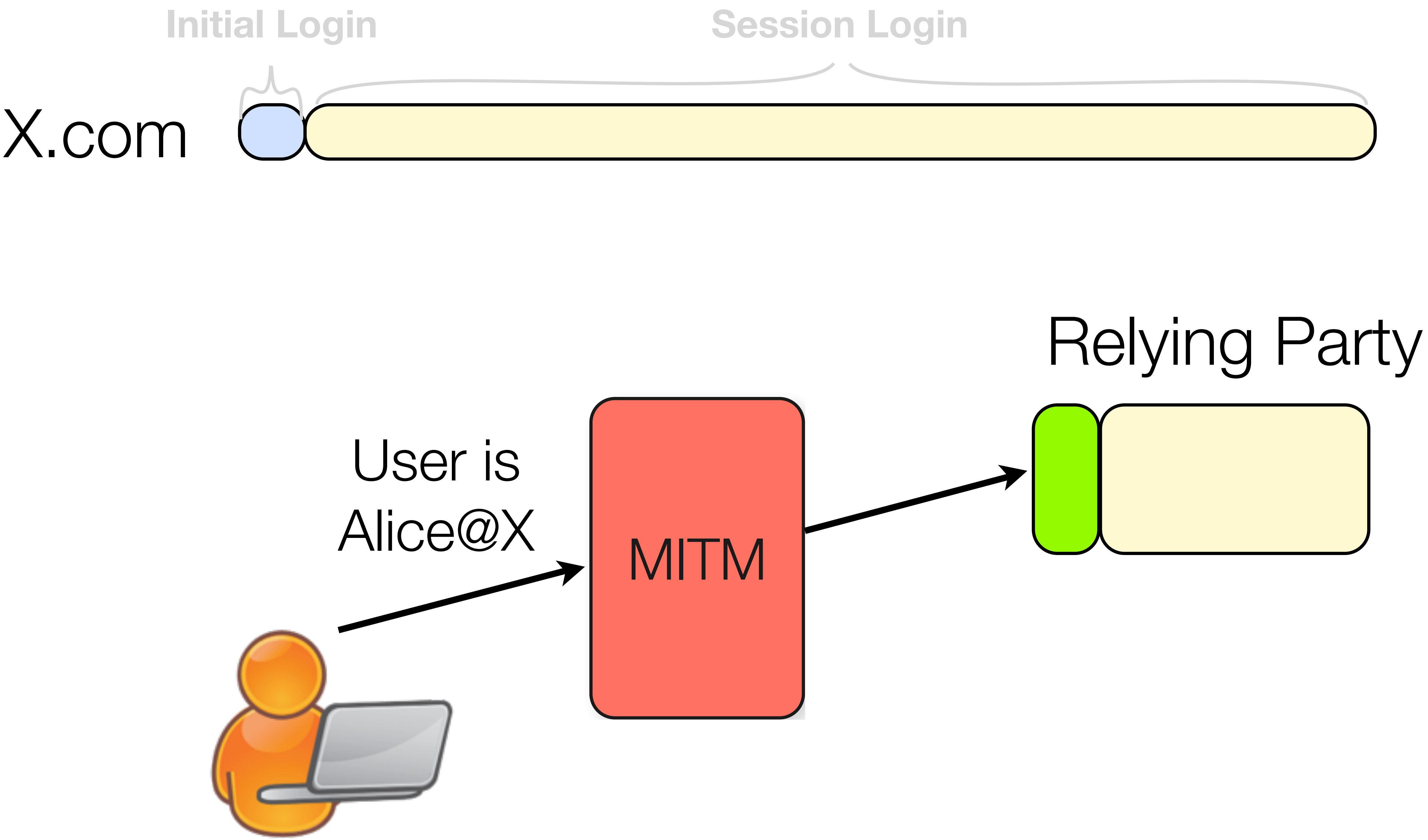
Session Login

X.com

Relying Party

User is
Alice@X

Identity assertion theft



Initial Login

Session Login

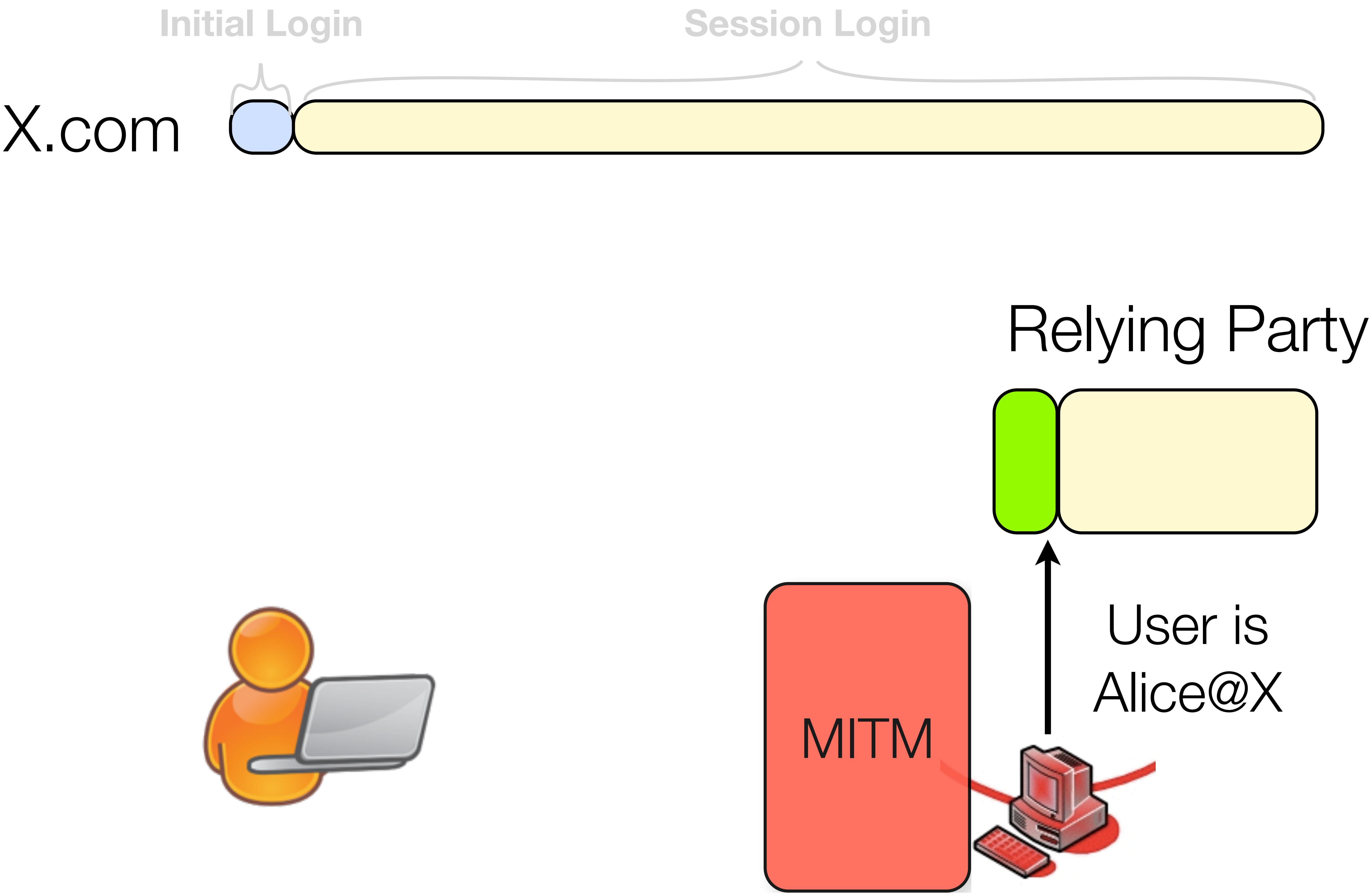
X.com

Relying Party

User is Alice@X

MITM

Identity assertion theft



Initial Login

Session Login

X.com

Relying Party

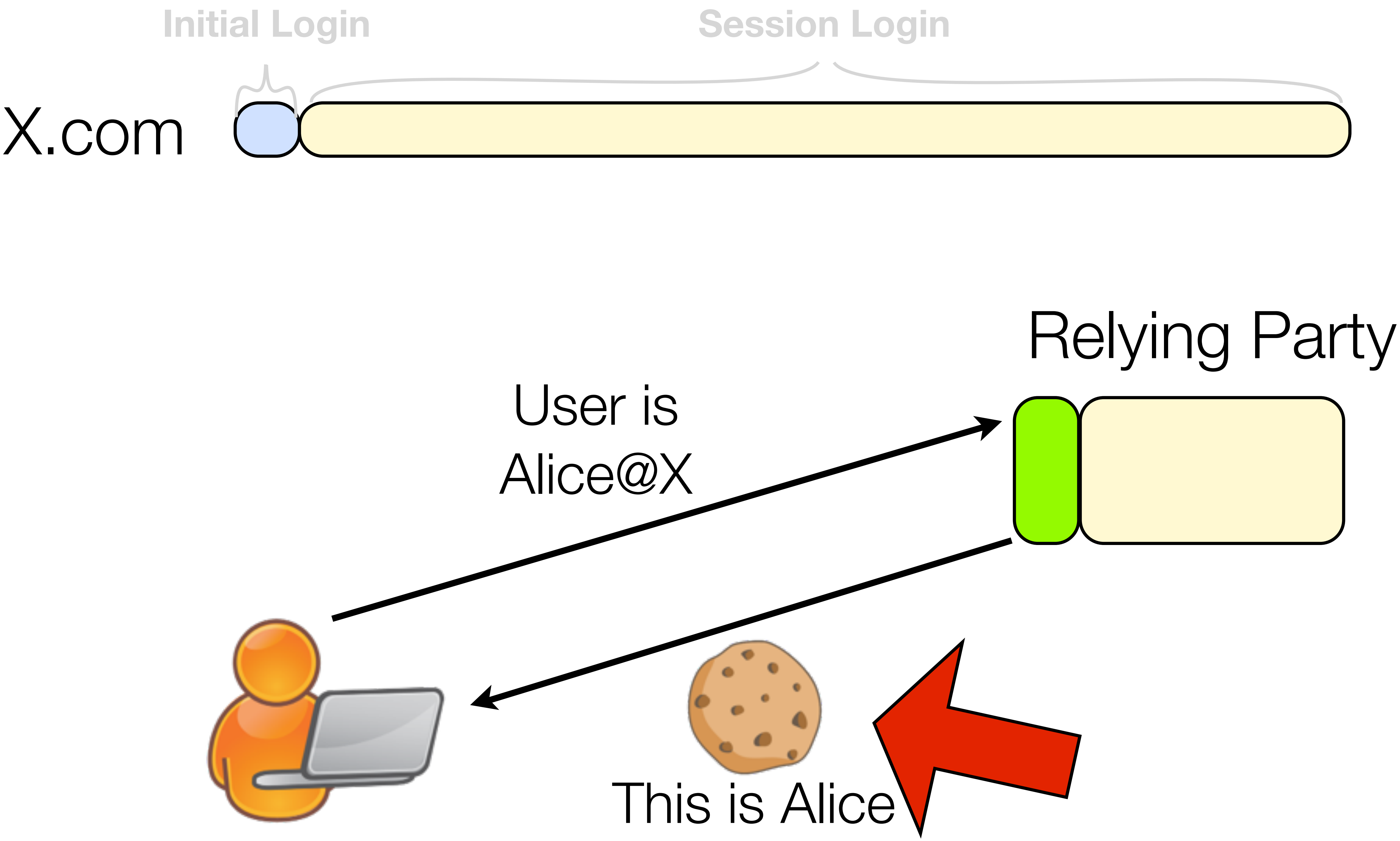
MITM

User is Alice@X

Two areas for attack

- MITM the connection between user and RP
  - Replay identity assertions
- Steal relying party cookie after login

RP cookie theft



Initial Login

Session Login

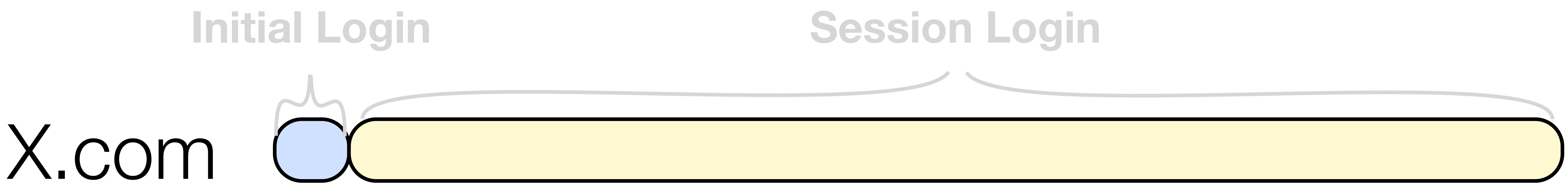
X.com

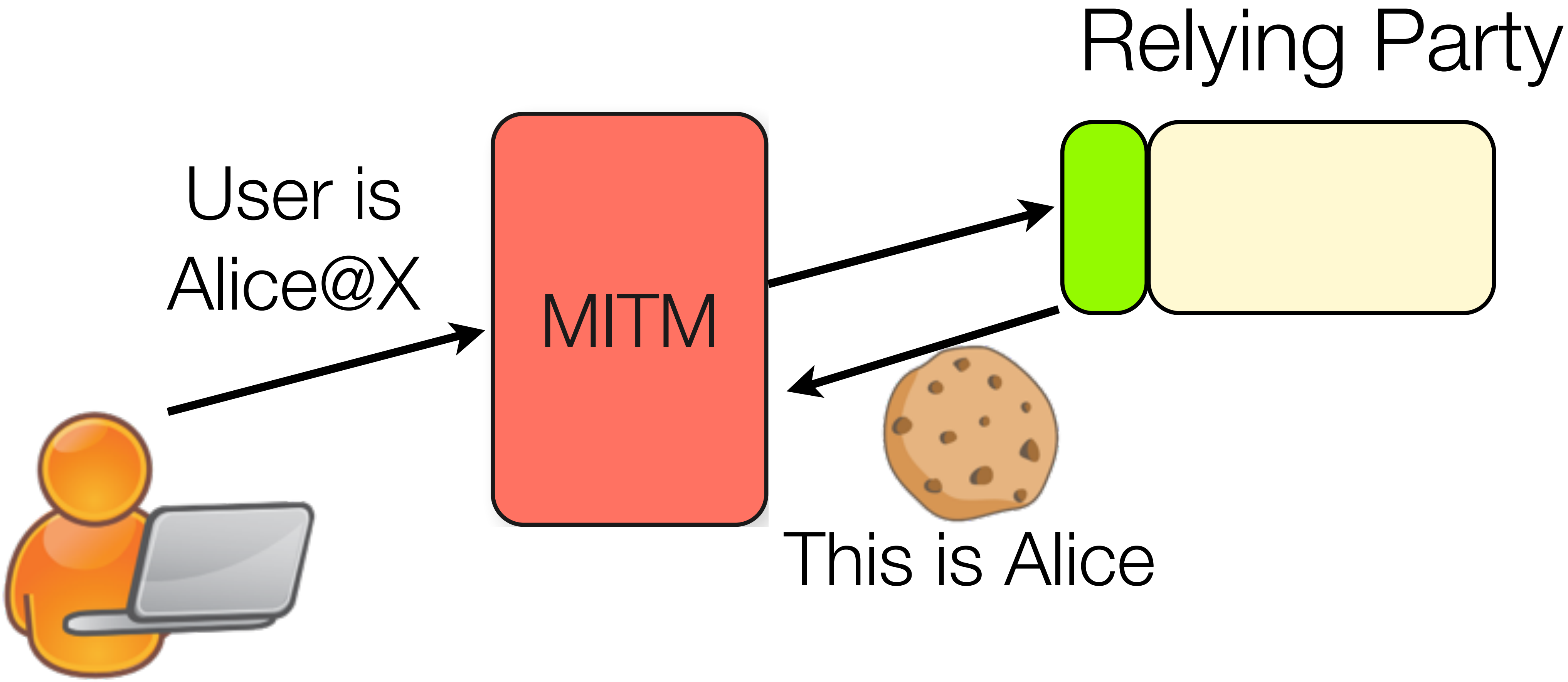
Relying Party

User is Alice@X

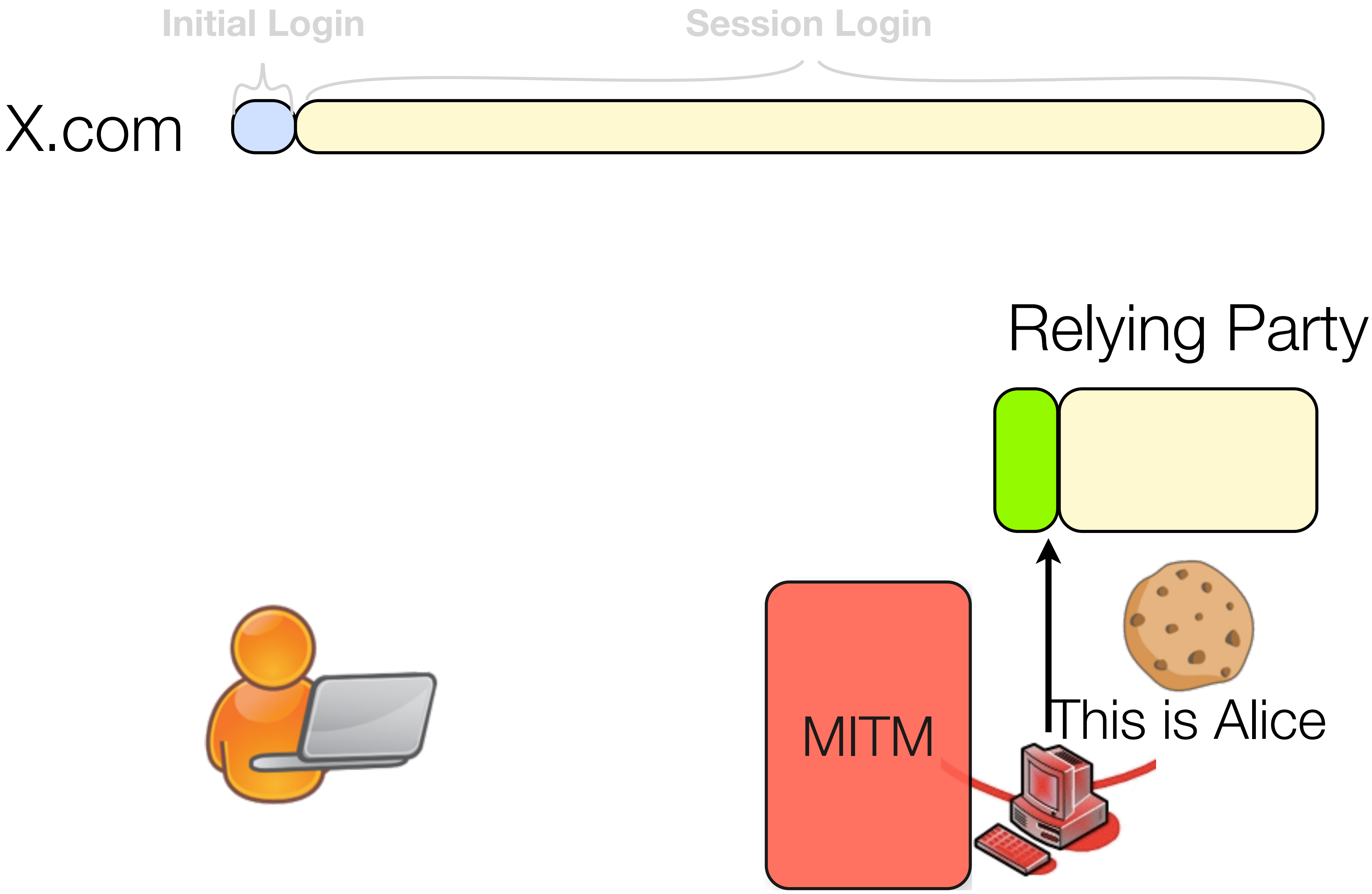
This is Alice

RP cookie theft





RP cookie theft



Initial Login

Session Login

X.com

Relying Party

MITM

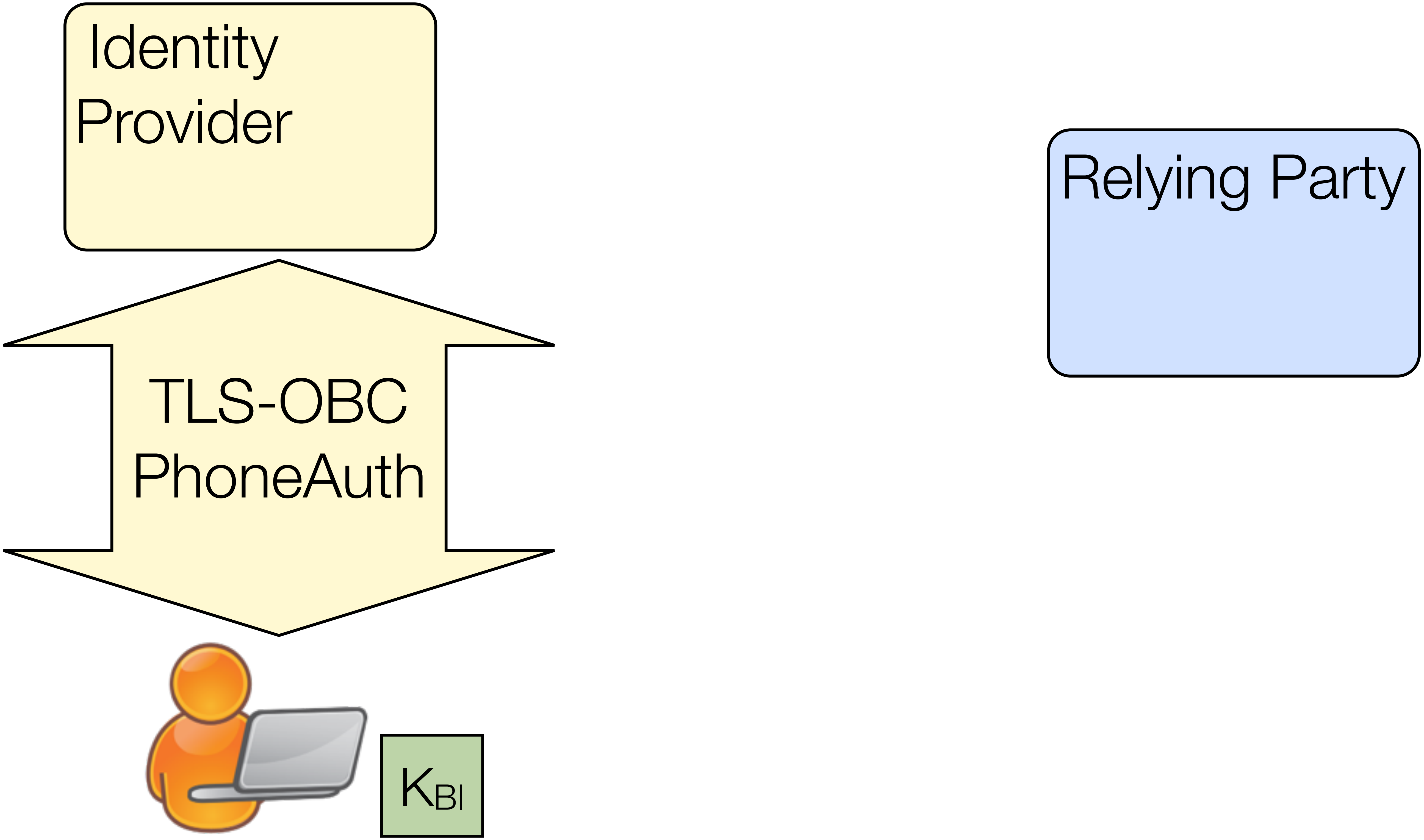
This is Alice

Extensions to Persona

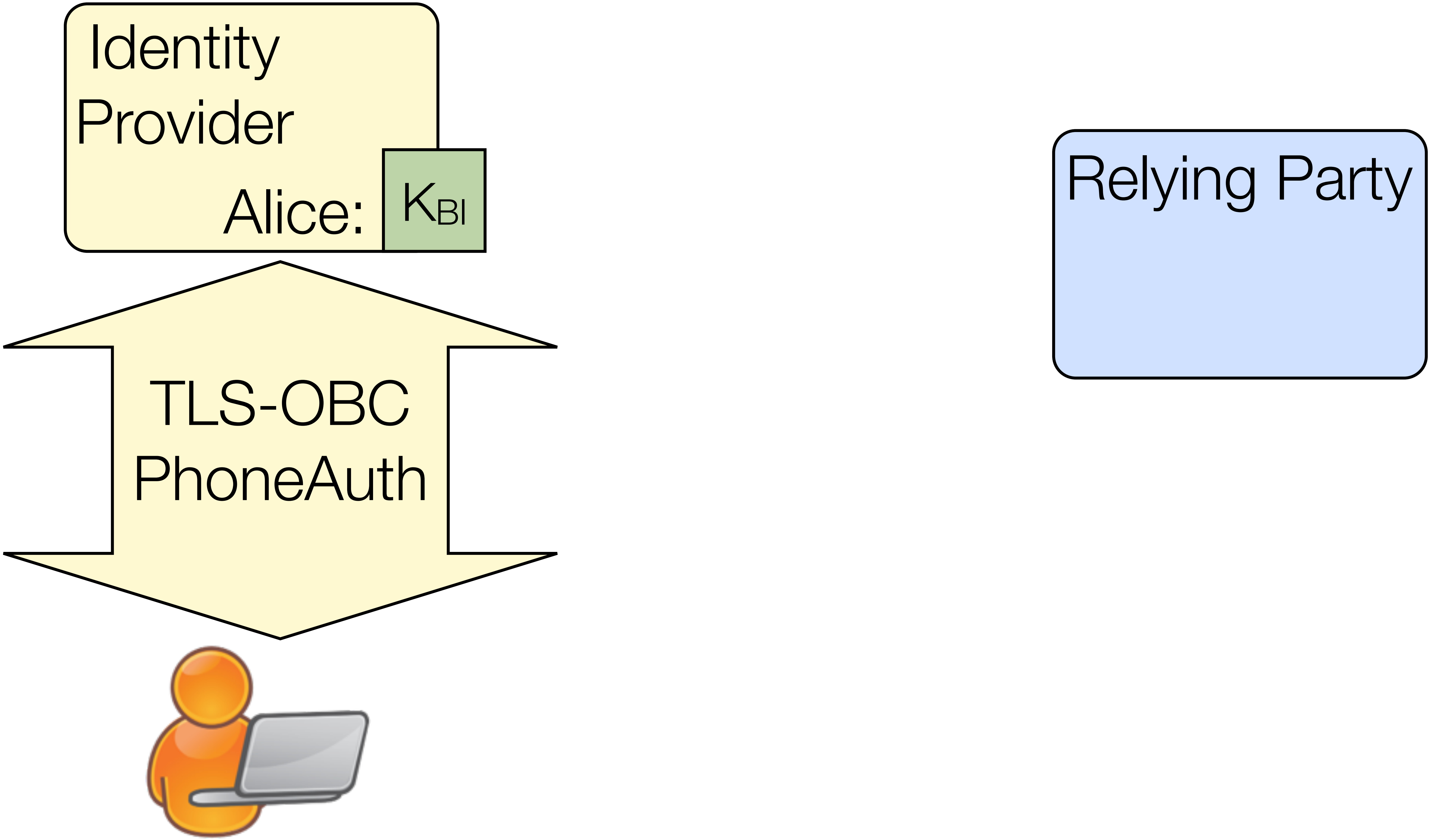
Design Goals

- Strengthen identity assertions against MITM theft
- Allow relying parties to establish a key for communication with the user

Initial Assumptions



Initial Assumptions



Identity Provider

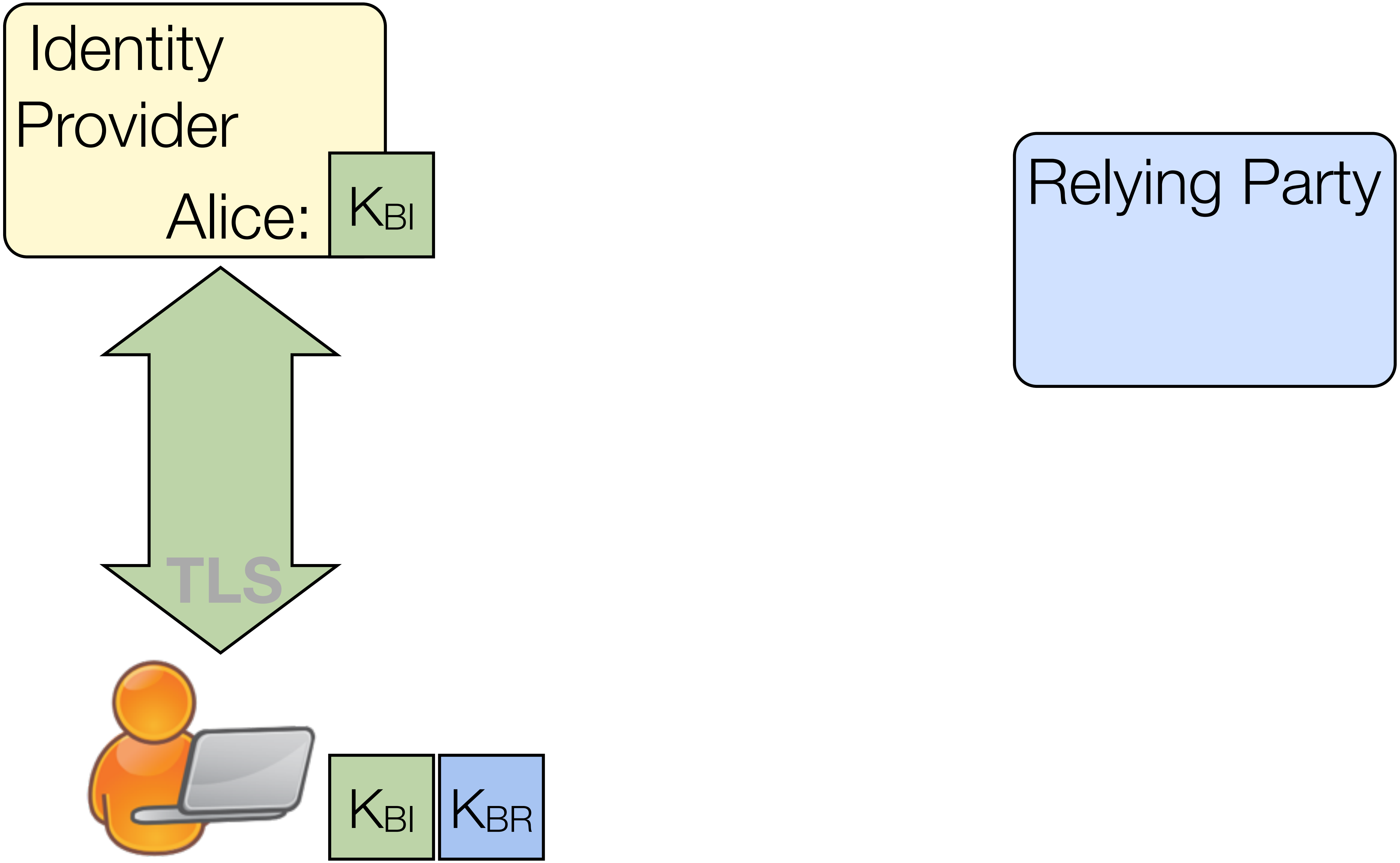
Alice: K_{BI}

TLS-OBC

PhoneAuth

Relying Party

Initial Assumptions



Identity Provider

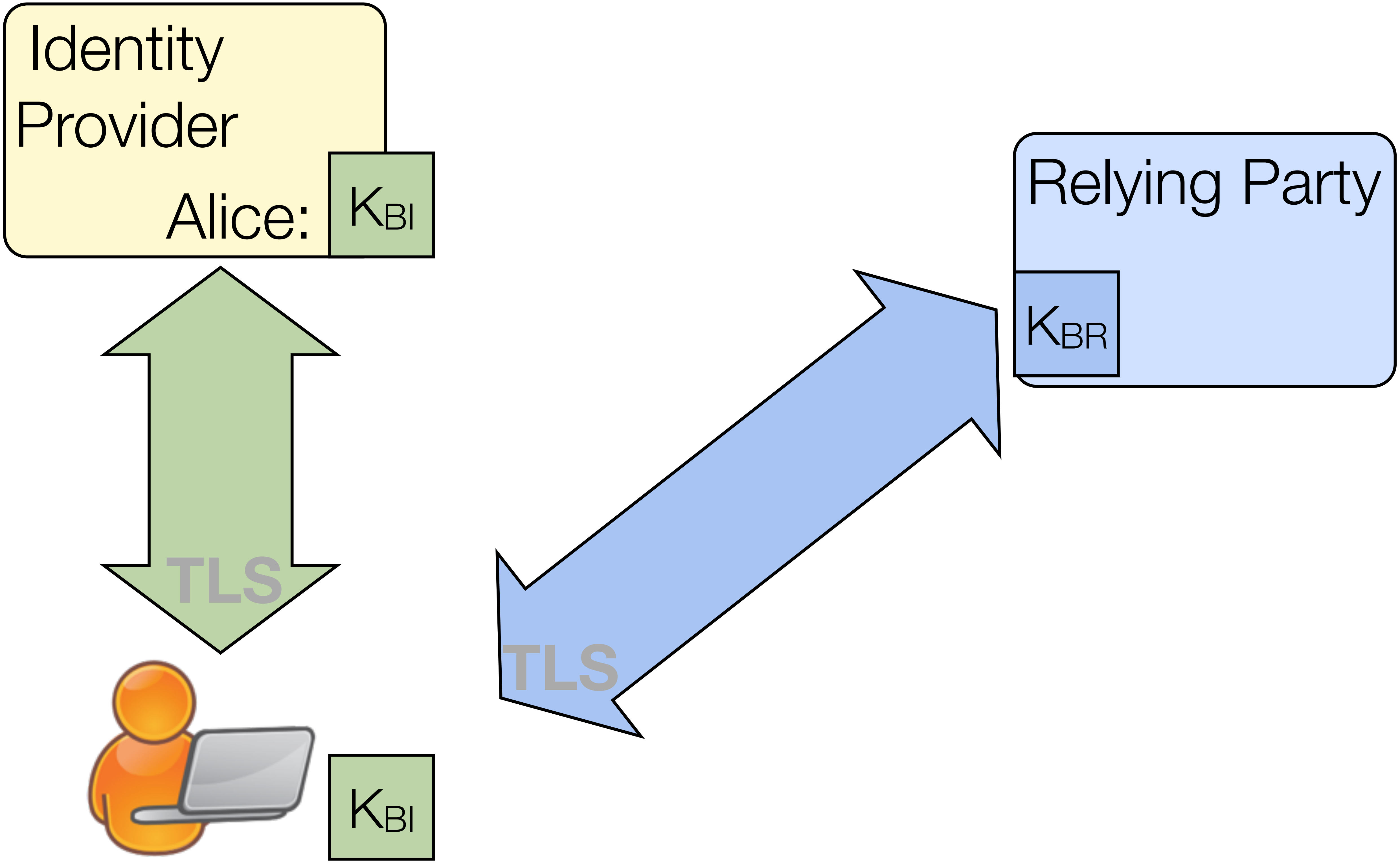
Alice: K_{BI}

TLS

K_{BI} K_{BR}

Relying Party

Initial Assumptions



Identity Provider

Alice: K_{BI}

Relying Party

K_{BR}

TLS

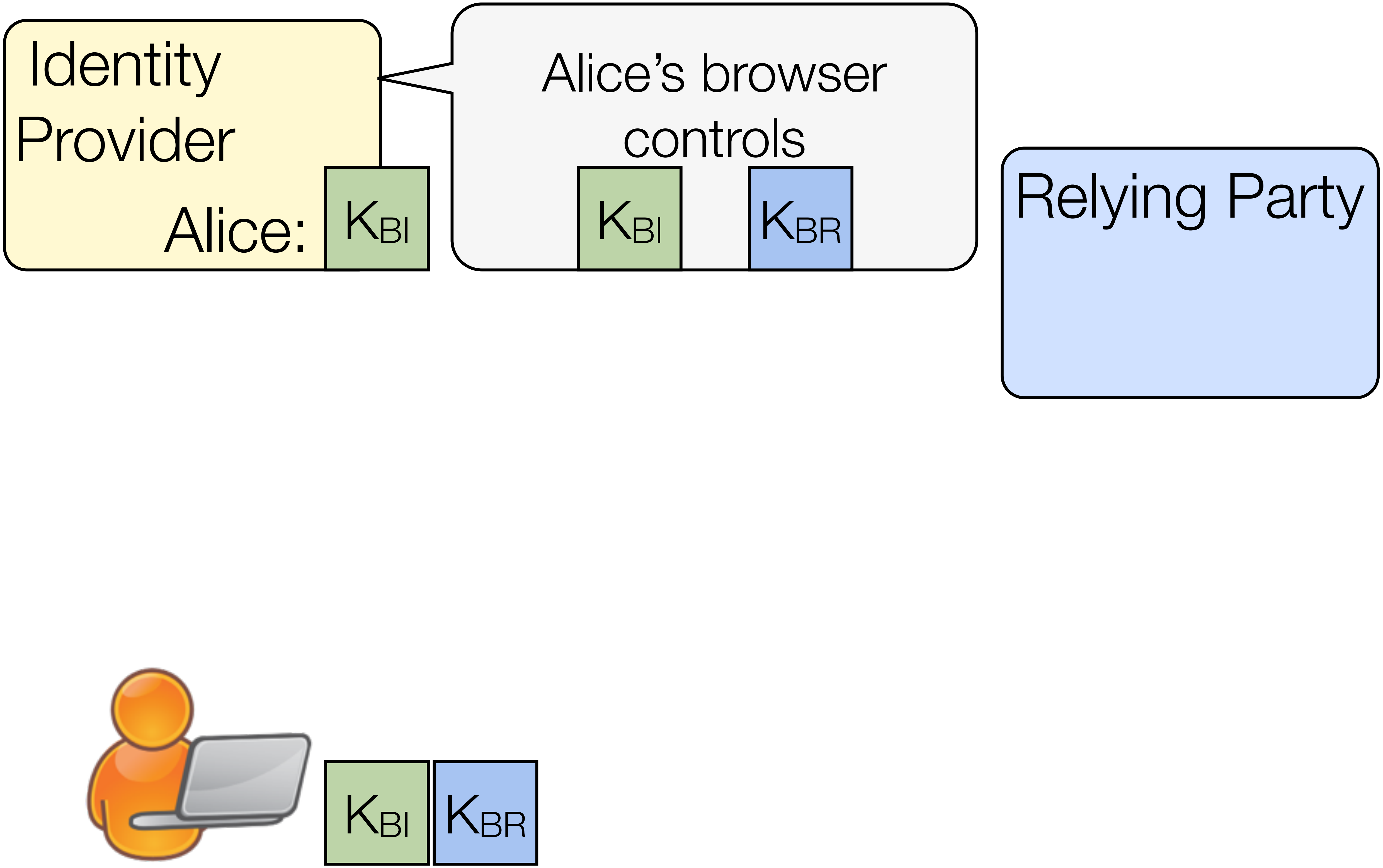
TLS

K_{BI}

Persona-OBC-Central

- Uses the Persona underpinnings, works more like OAuth2
  - IDP sees RP’s public key
  - Can track user logins to RPs
  - Simple to implement

Goal



Identity Provider

Alice: K_{BI}

Alice's browser controls

K_{BI} K_{BR}

Relying Party

K_{BI} K_{BR}

Post Key API

Post Key API

- **Goal**: Convince IDP that browser controls two OBCs used on two different domains

Post Key API

- **Goal**: Convince IDP that browser controls two OBCs used on two different domains
- Creates cross certification between two origin bound certificate keys

Post Key API

- **Goal**: Convince IDP that browser controls two OBCs used on two different domains

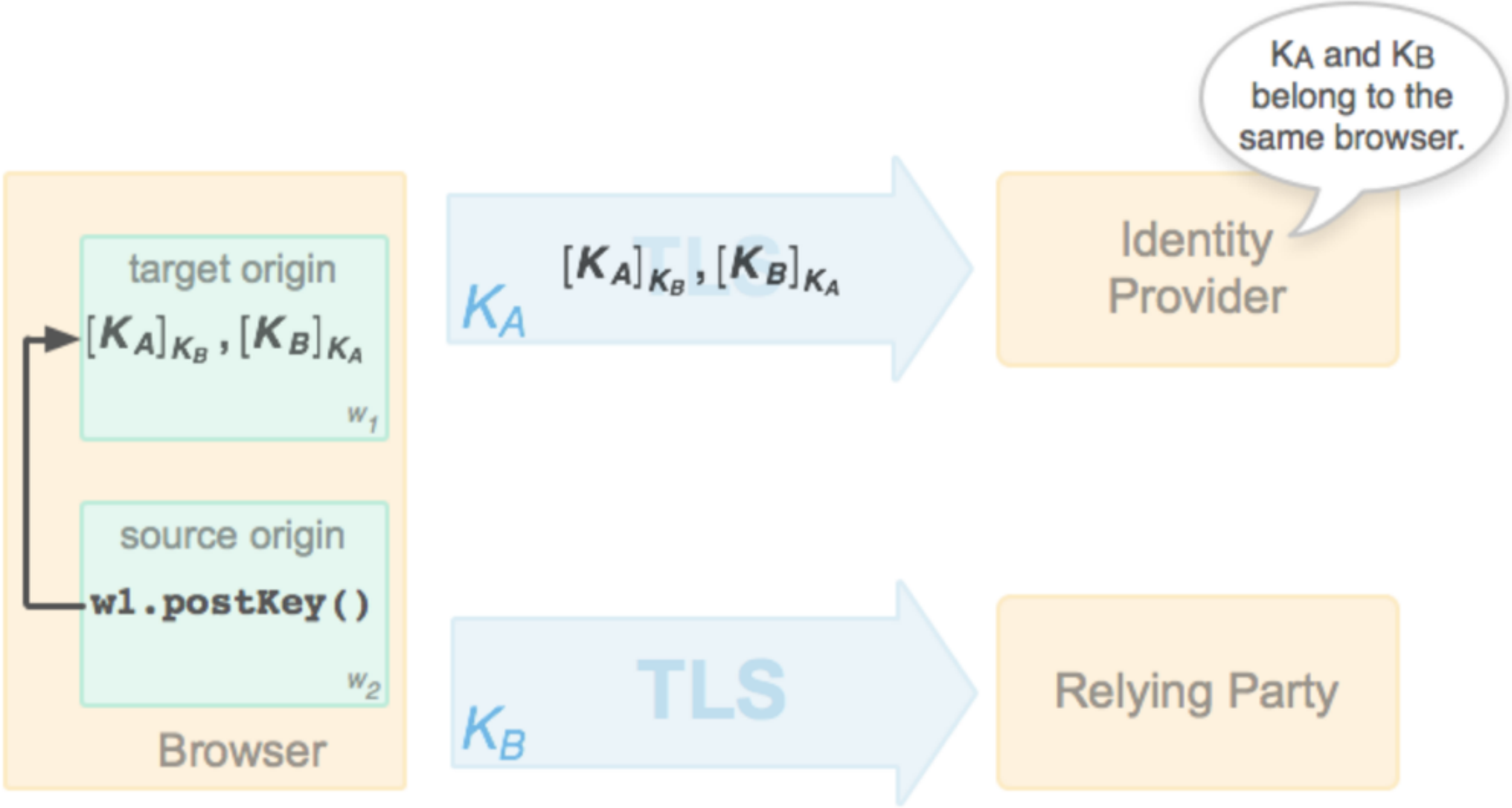
- Creates cross certification between two origin bound certificate keys
- API exposed as browser extension

Post Key API

- **Goal**: Convince IDP that browser controls two OBCs used on two different domains

- Creates cross certification between two origin bound certificate keys
- API exposed as browser extension
    - Similar to postMessage() call

Post Key API



K_A and K_B belong to the same browser.

Post Key API

- Assumptions
  - IDP received cross cert on TLS channel associated with K_A
  - IDP knows K_A is a key Alice's browser controls

Post Key API

- Assumptions
  - IDP received cross cert on TLS channel associated with K_A
  - IDP knows K_A is a key Alice's browser controls

$$[K_A, \text{A.com}]_{KB} \text{ , } [K_B, \text{B.com}]_{KA}$$

Post Key API

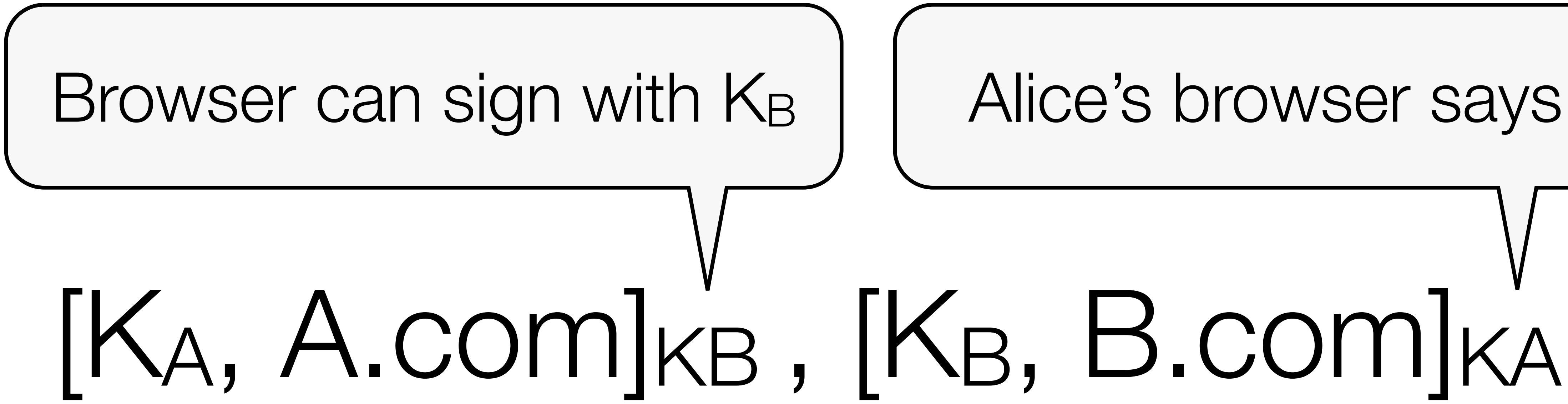
- Assumptions
  - IDP received cross cert on TLS channel associated with K_A
  - IDP knows K_A is a key Alice's browser controls

Alice's browser says K_B

$$[K_A, \text{A.com}]_{KB} , [K_B, \text{B.com}]_{KA}$$

Post Key API

- Assumptions
  - IDP received cross cert on TLS channel associated with K_A
  - IDP knows K_A is a key Alice's browser controls

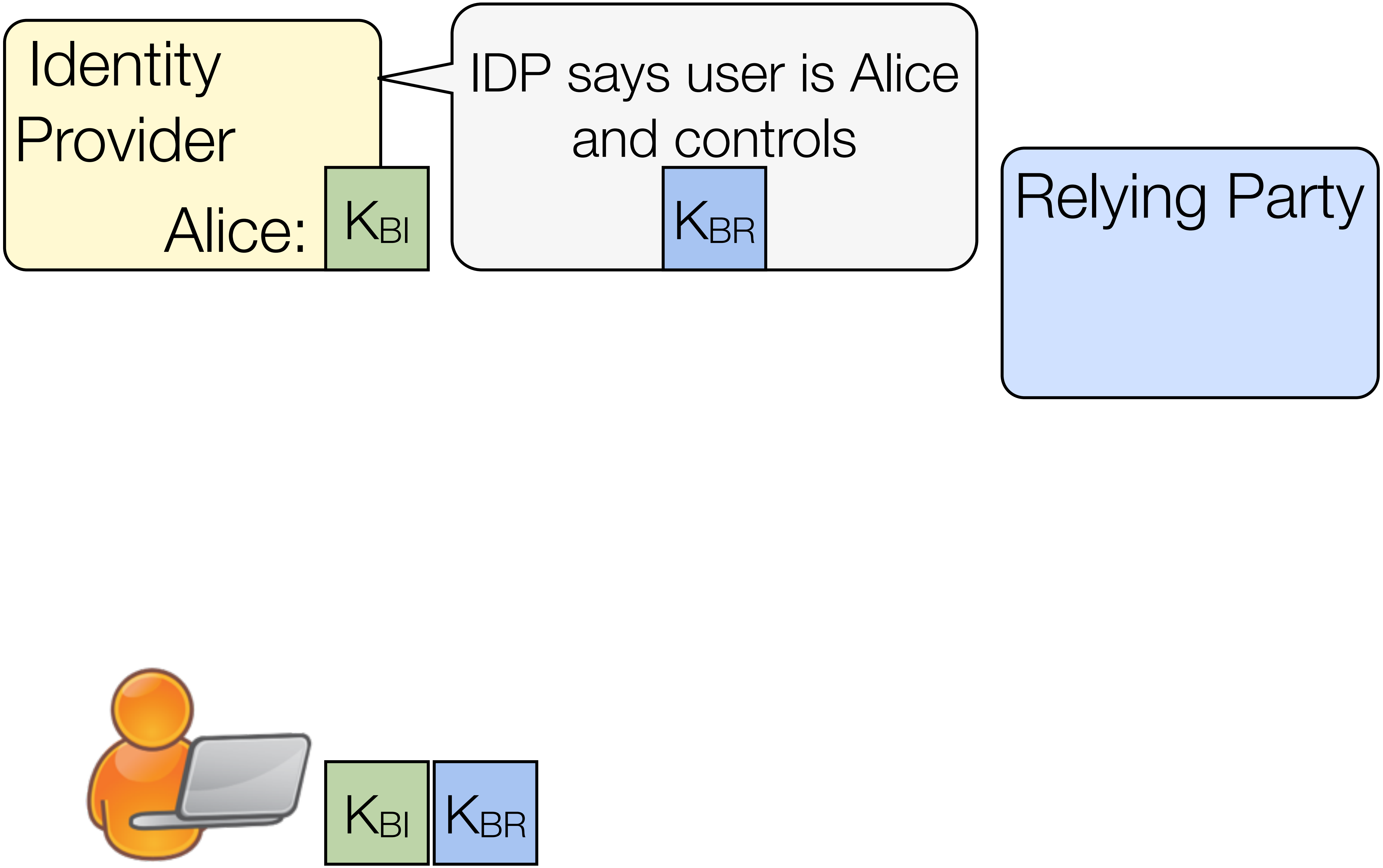


Browser can sign with K_B

Alice's browser says K_B

$$[K_A, \text{A.com}]_{KB} , [K_B, \text{B.com}]_{KA}$$

Goal



Identity Provider

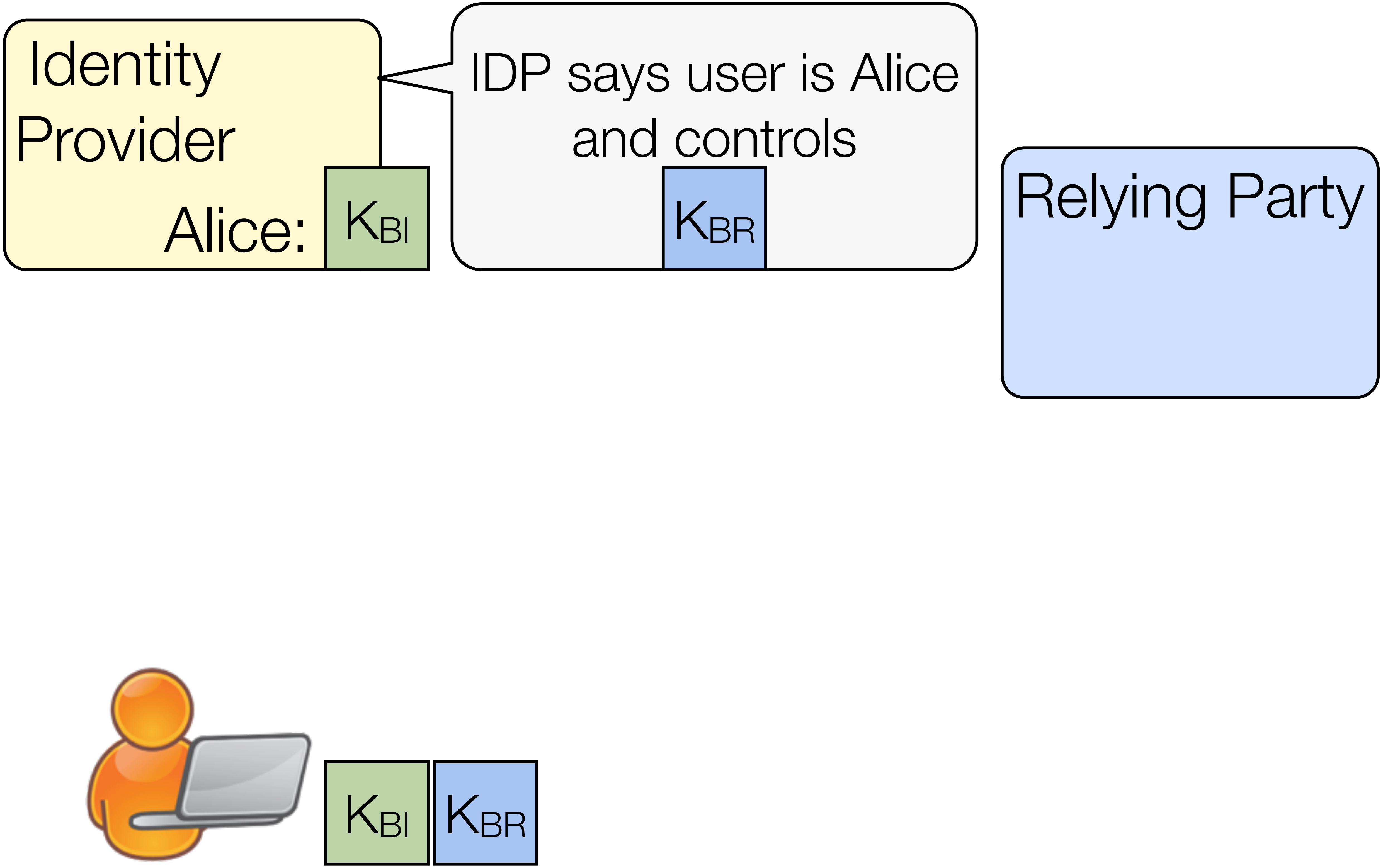
Alice: K_{BI}

IDP says user is Alice and controls K_{BR}

Relying Party

K_{BI} K_{BR}

Goal



Identity Provider

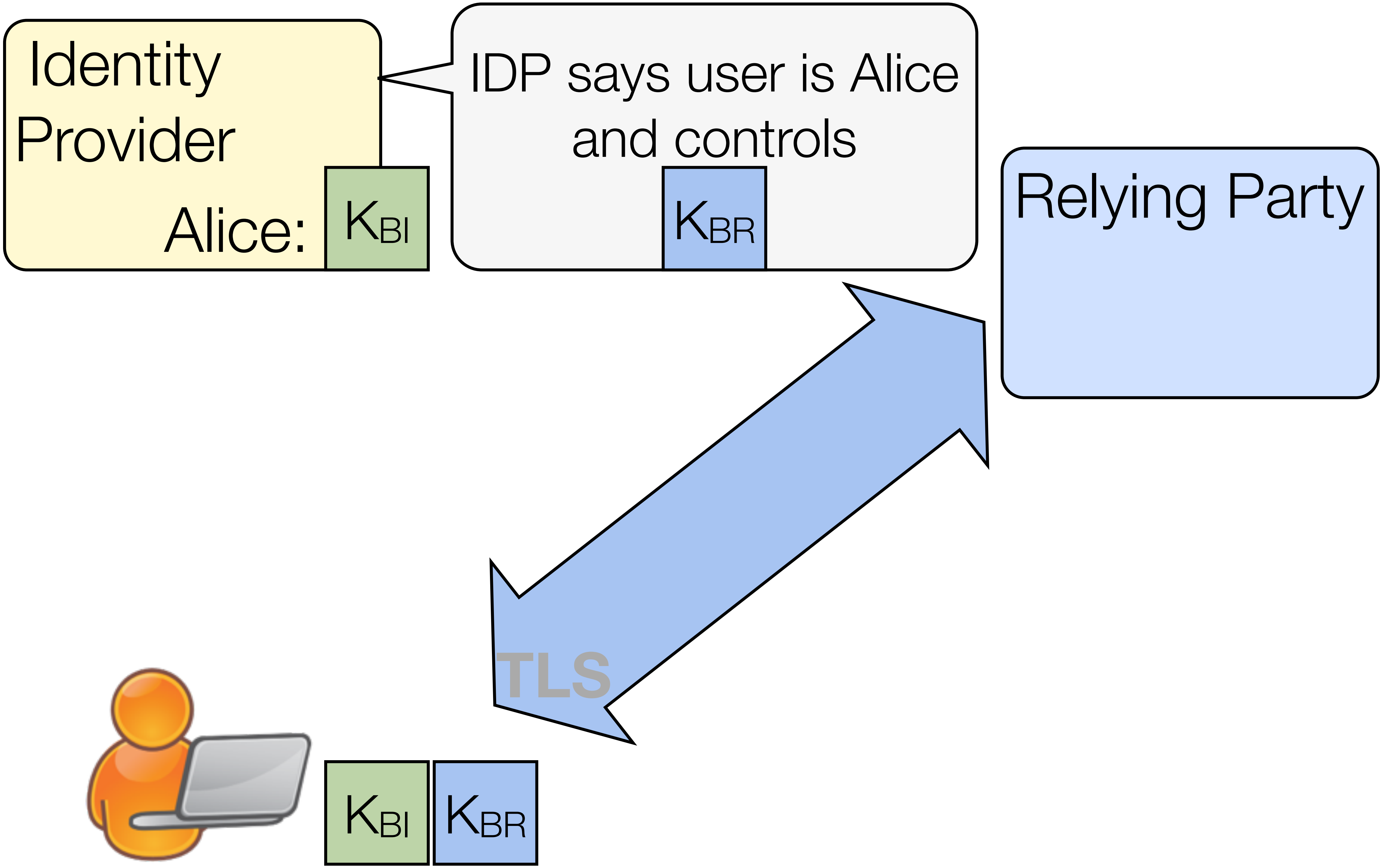
Alice: K_{BI}

IDP says user is Alice and controls K_{BR}

Relying Party

K_{BI} K_{BR}

Goal



Identity Provider

Alice: K_{BI}

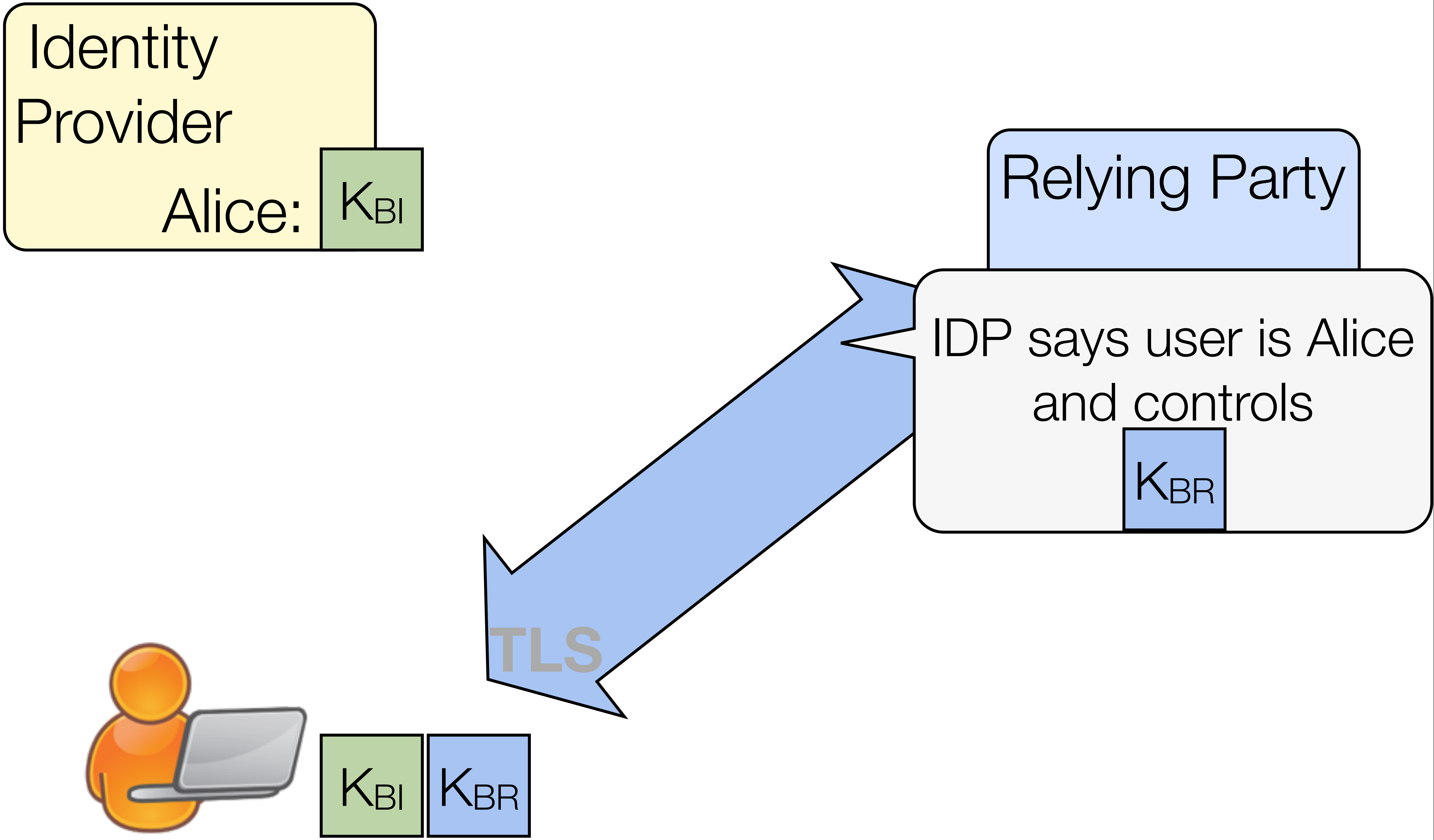
IDP says user is Alice and controls K_{BR}

Relying Party

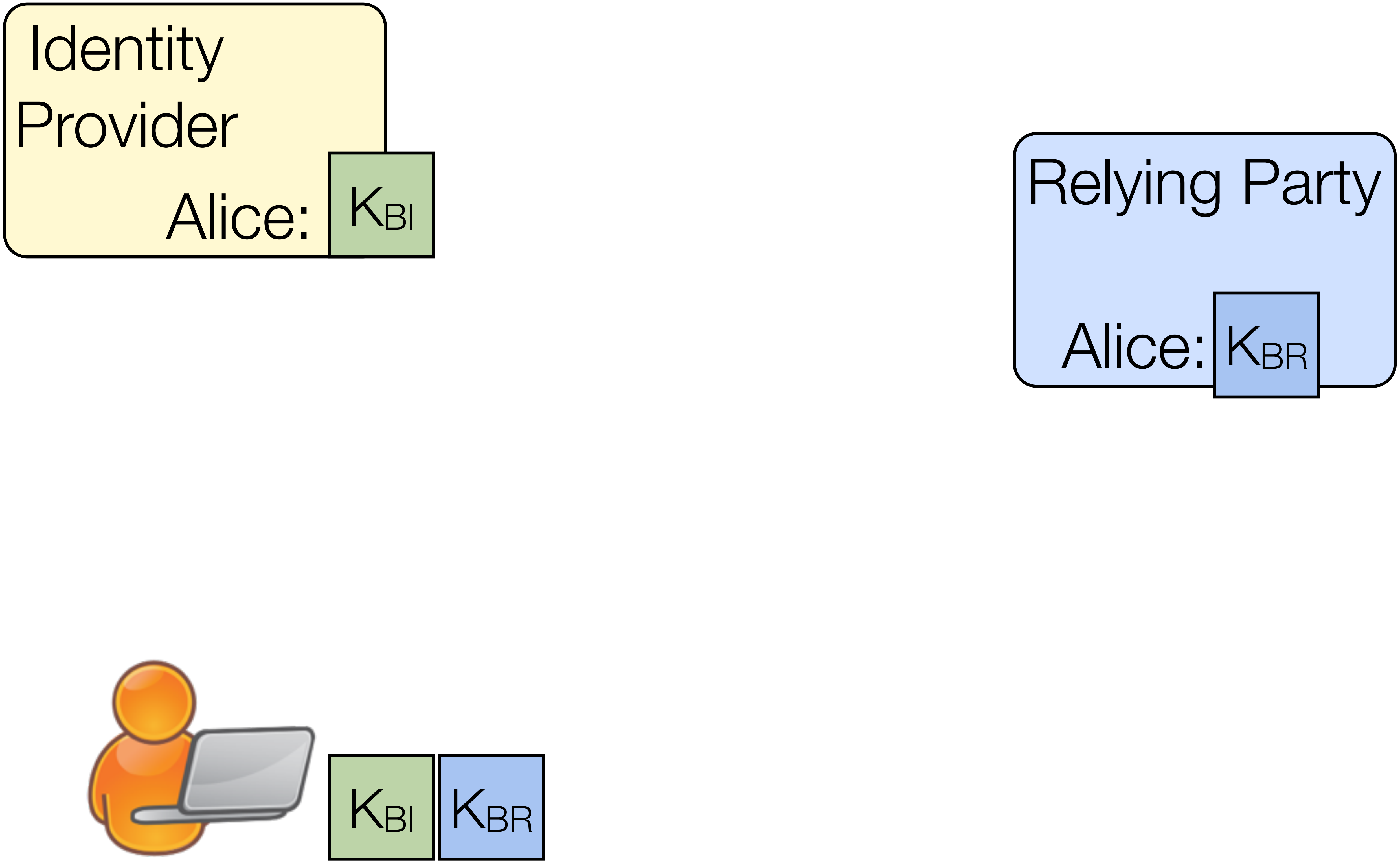
TLS

K_{BI} K_{BR}

Goal


Identity Provider
Alice: K_{BI}
Relying Party
IDP says user is Alice and controls K_{BR}
TLS
K_{BI}
K_{BR}

Goal



Identity Provider

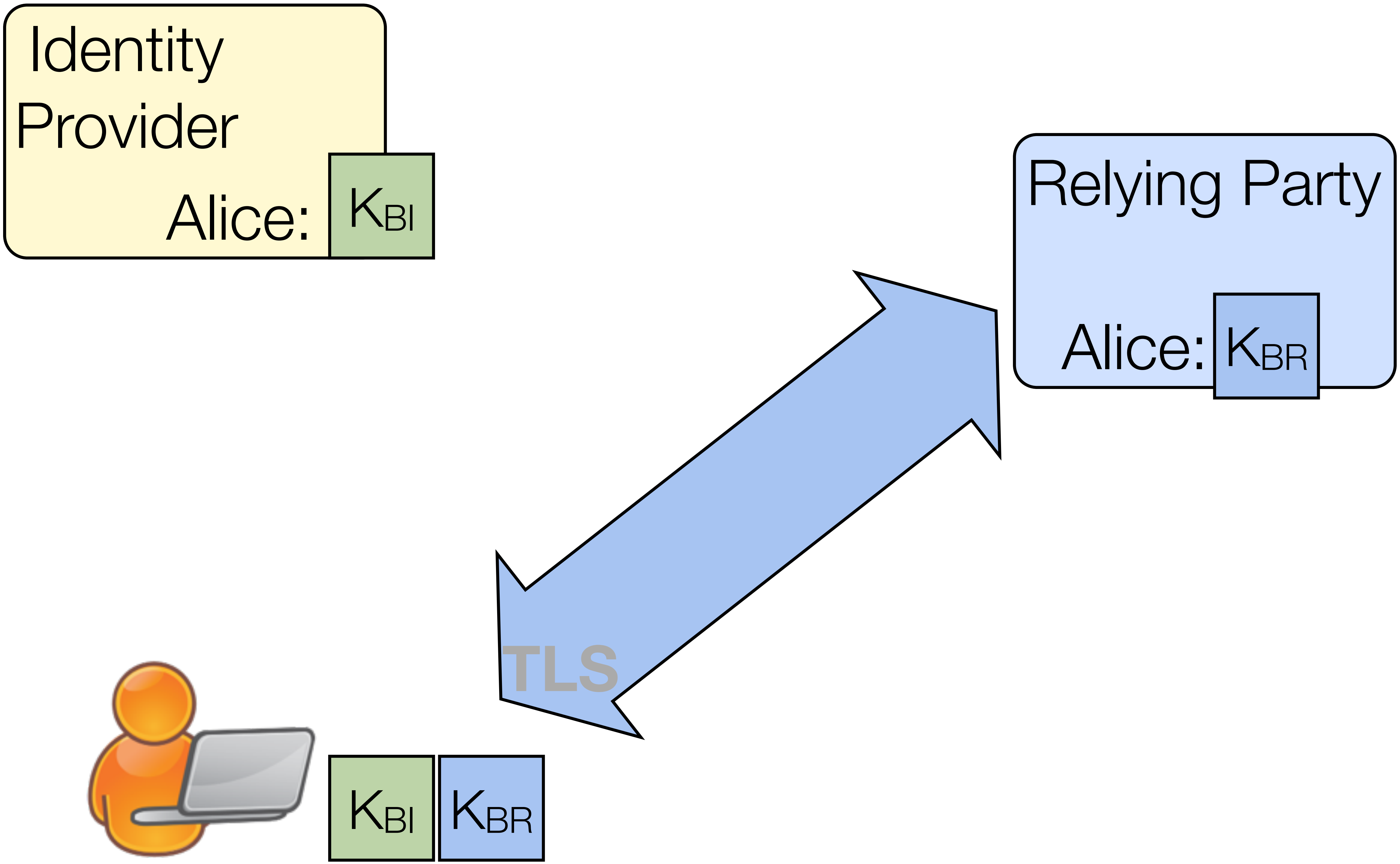
Alice: K_{BI}

Relying Party

Alice: K_{BR}

K_{BI} K_{BR}

Goal



Identity Provider

Alice: K_{BI}

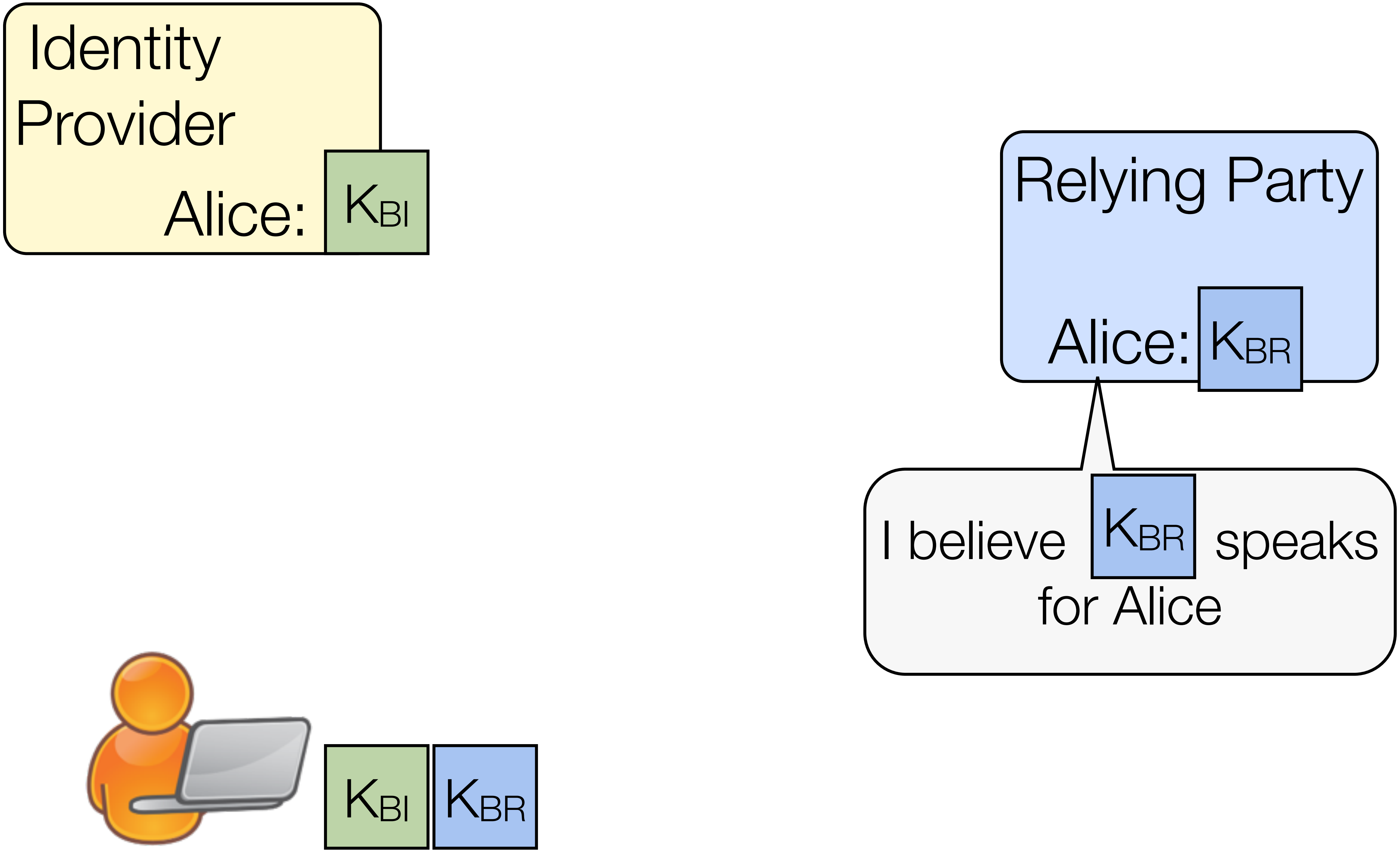
Relying Party

Alice: K_{BR}

TLS

K_{BI} K_{BR}

Goal



Identity Provider

Alice: K_{BI}

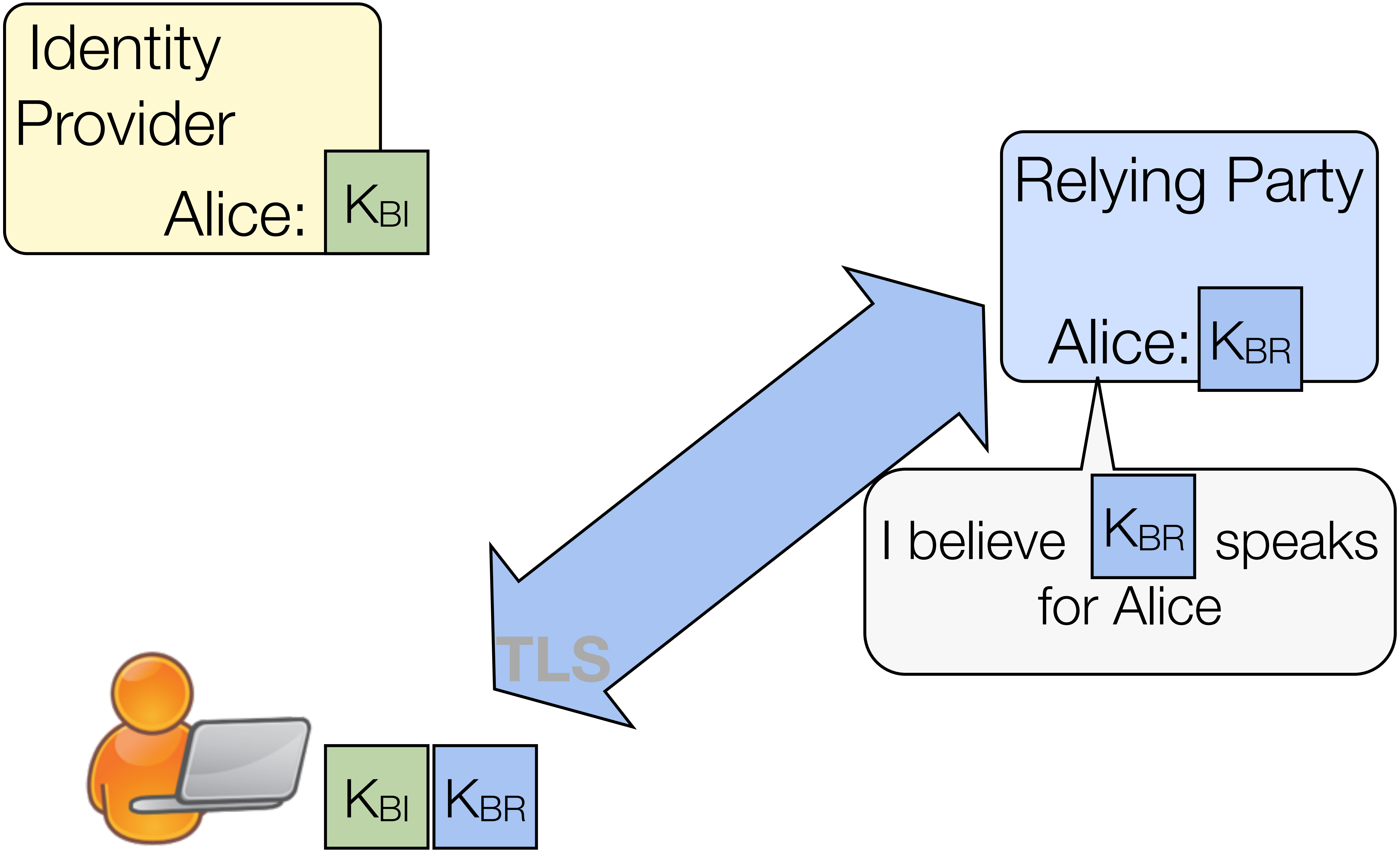
Relying Party

Alice: K_{BR}

I believe K_{BR} speaks for Alice

K_{BI} K_{BR}

Goal


Identity Provider
Alice: K_{BI}
Relying Party
Alice: K_{BR}
I believe K_{BR} speaks for Alice
TLS
K_{BI} K_{BR}

Persona-OBC-Local:

Preserve Persona semantics

Persona Specifics

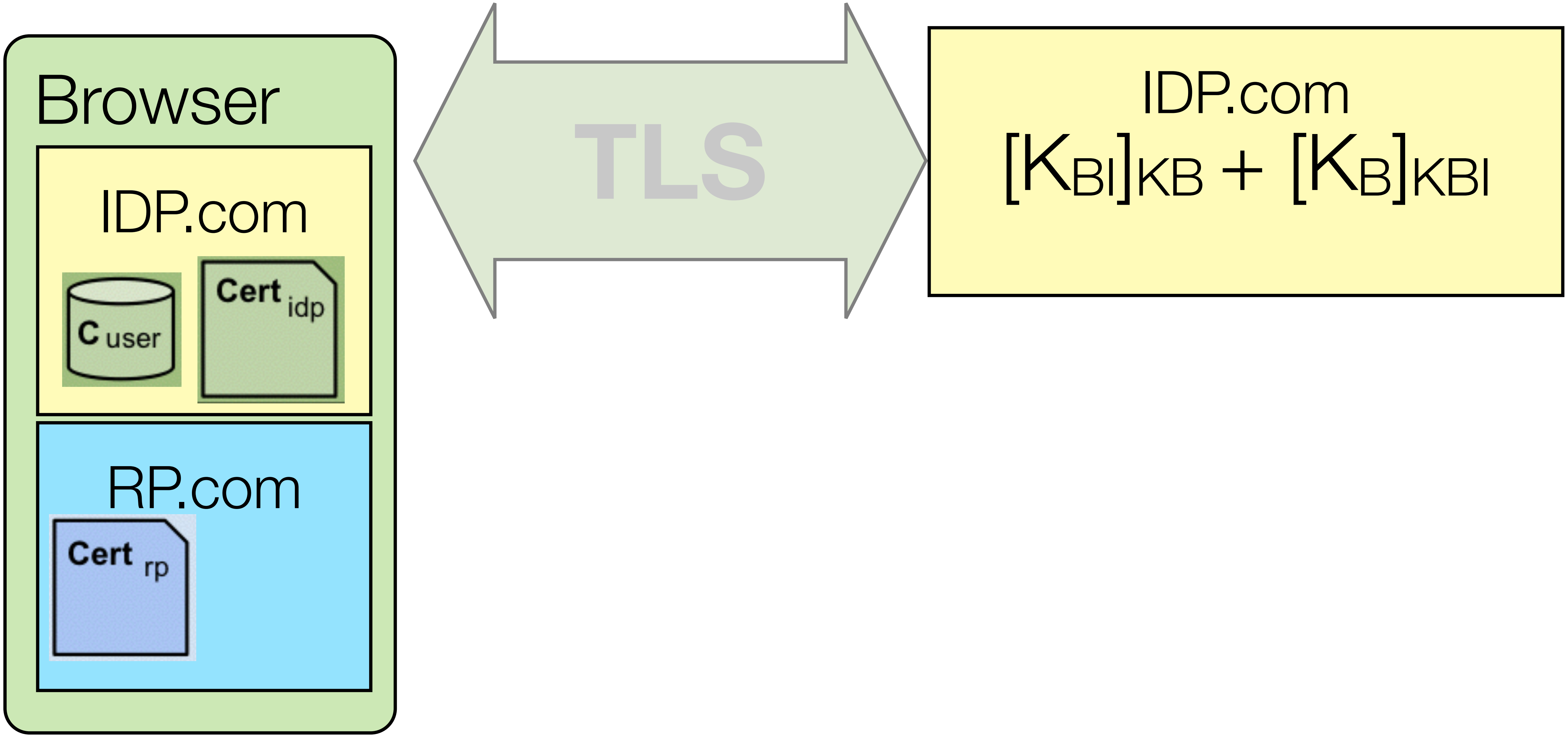
- IDP cannot track where the user logs in
- Uses public key crypto (in the browser)
  - IDP signs short lived browser key
  - Browser creates identity assertion with browser key
  - RP can verify assertions without an online IDP

Persona-OBC-Local

- IDP signs browser controlled key K_B and user identity with its well known key K_I
- Browser creates identity assertions on the fly by signing new TLS-OBC key for RP with K_B

Persona-OBC-Local protocol

1. Browser sends cross certification and channel bound cookie to IDP



Browser

IDP.com

C_{user}

$Cert_{idp}$

RP.com

$Cert_{rp}$

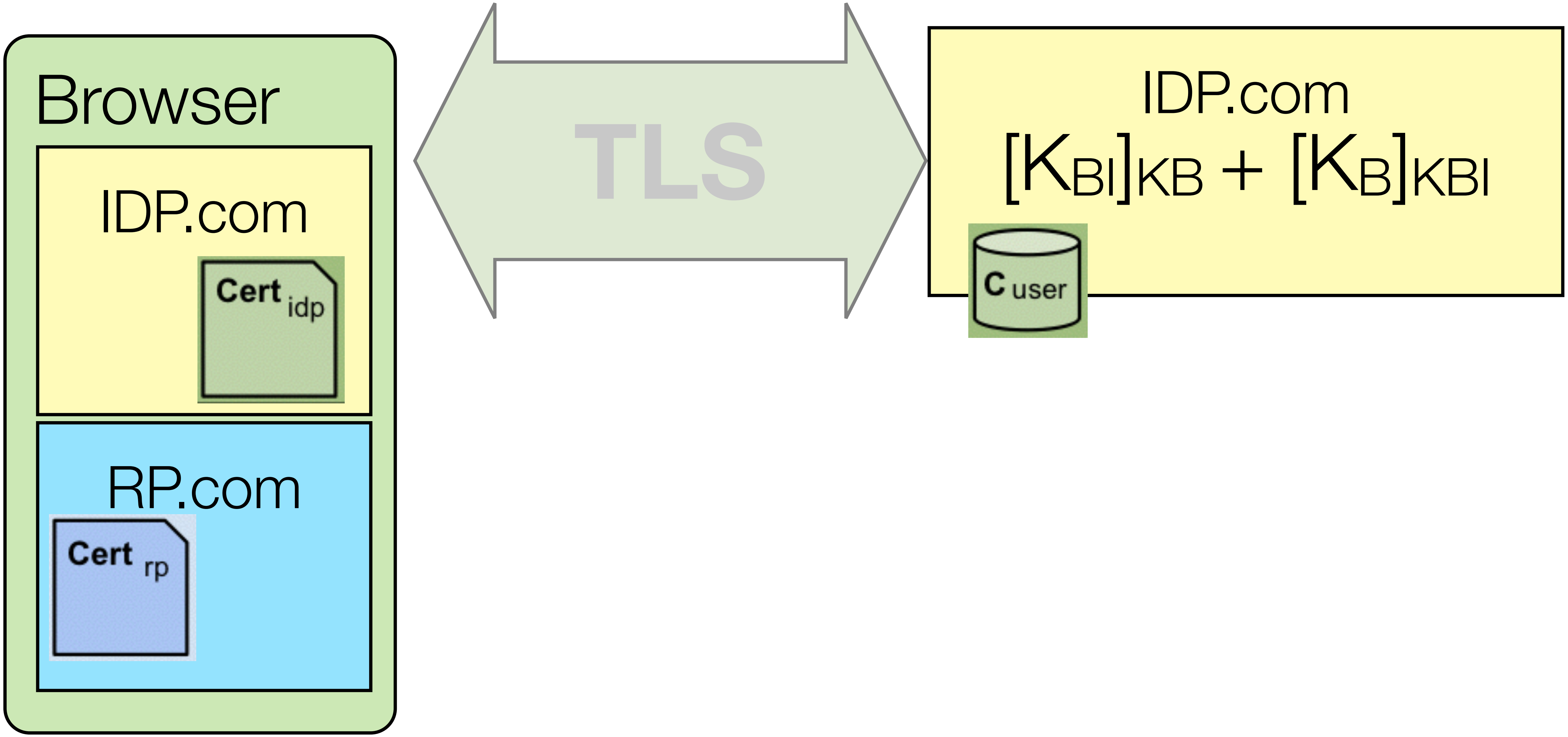
TLS

IDP.com

$[K_{BI}]_{KB} + [K_{B}]_{KBI}$

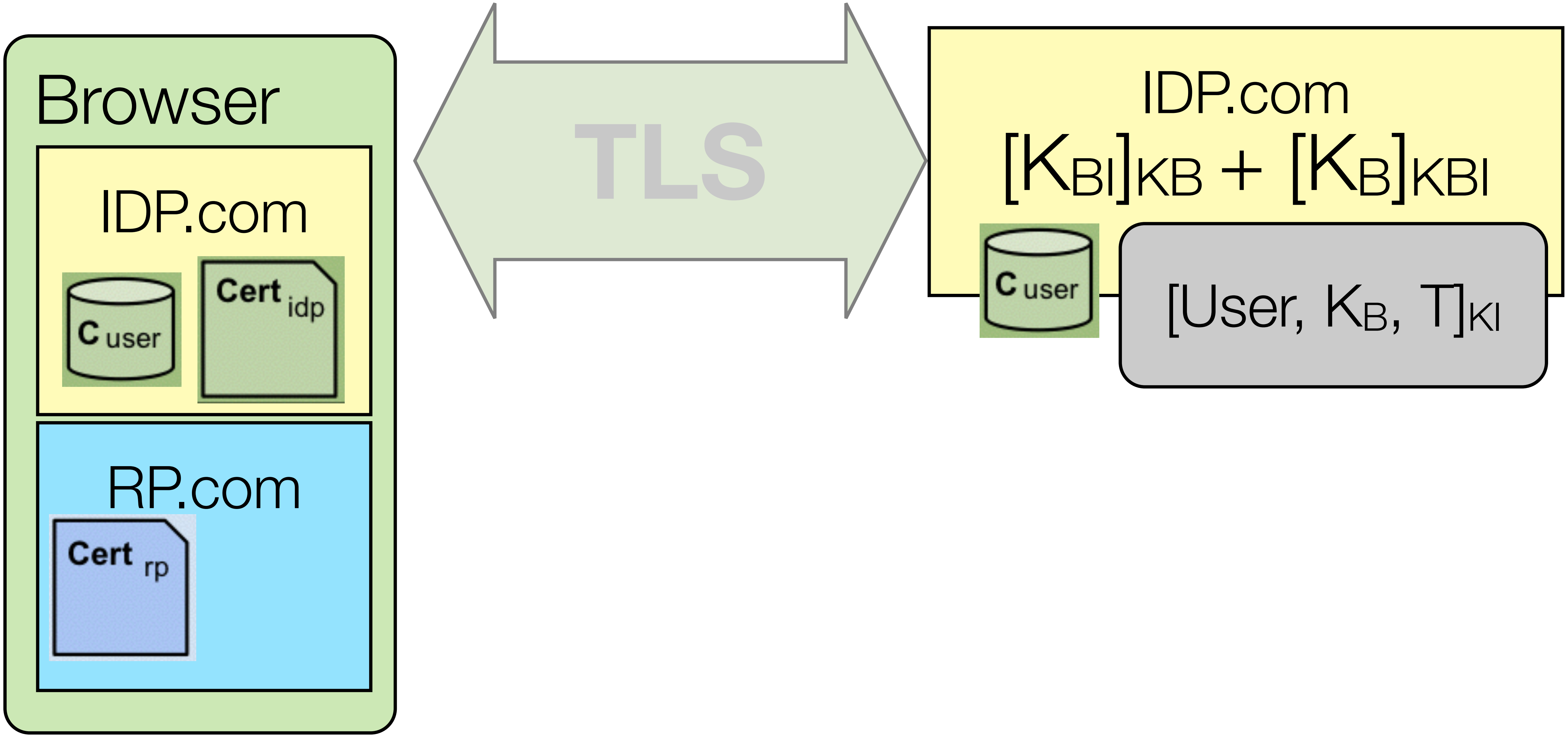
Persona-OBC-Local protocol

1. Browser sends cross certification and channel bound cookie to IDP



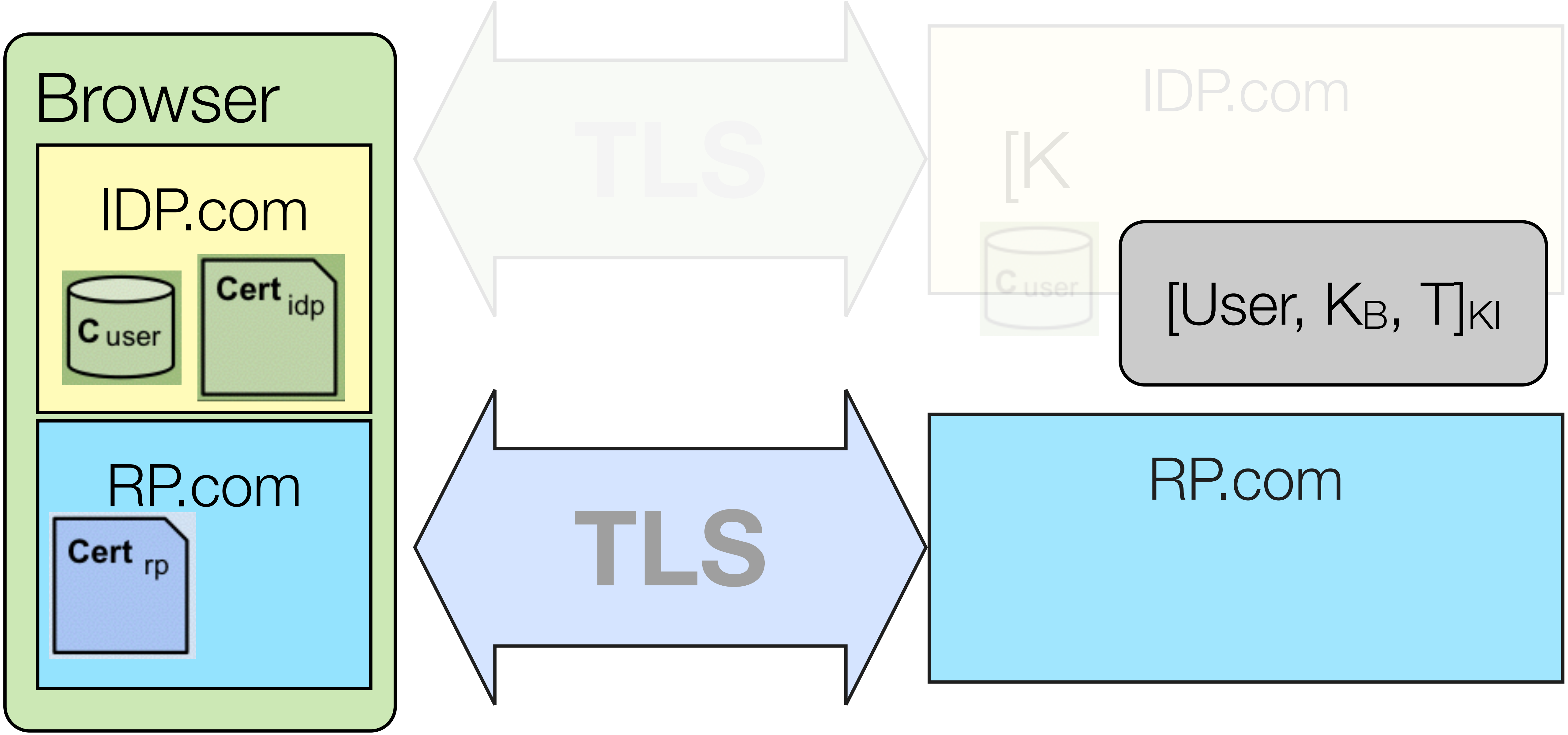
Persona-OBC-Local protocol

2. IDP creates identity certificate



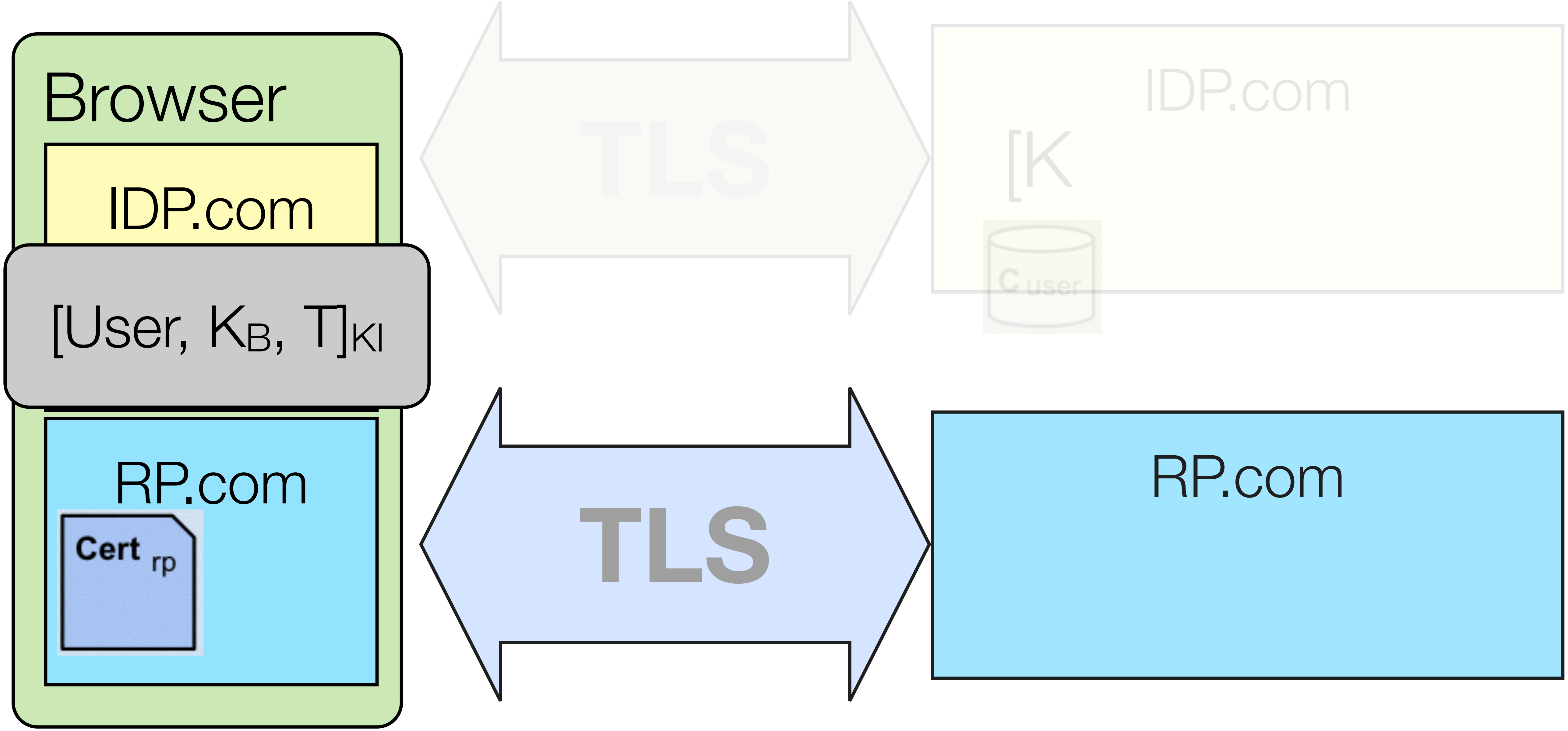
Persona-OBC-Local protocol

3. IDP sends identity certificate to browser for storage



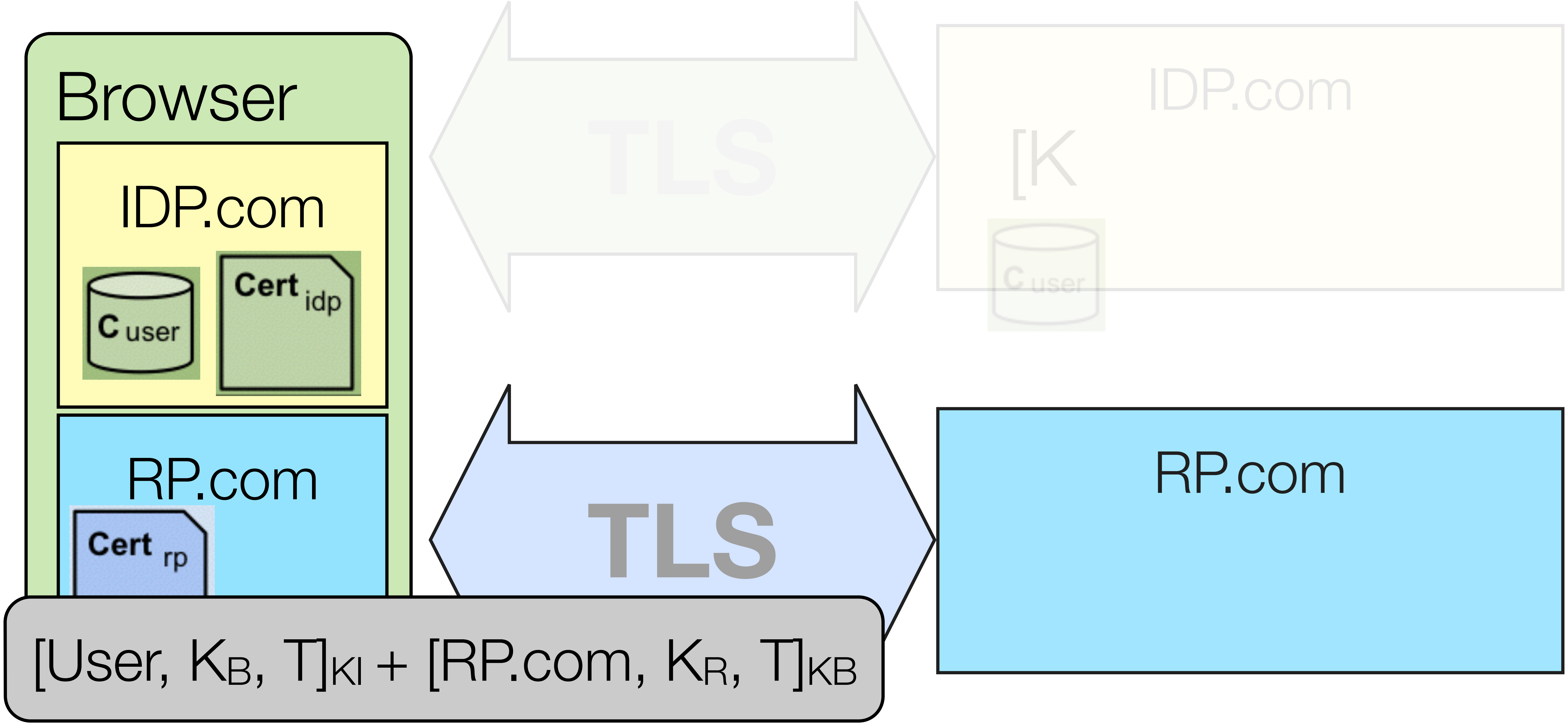
Persona-OBC-Local protocol

3. IDP sends identity certificate to browser for storage



Persona-OBC-Local protocol

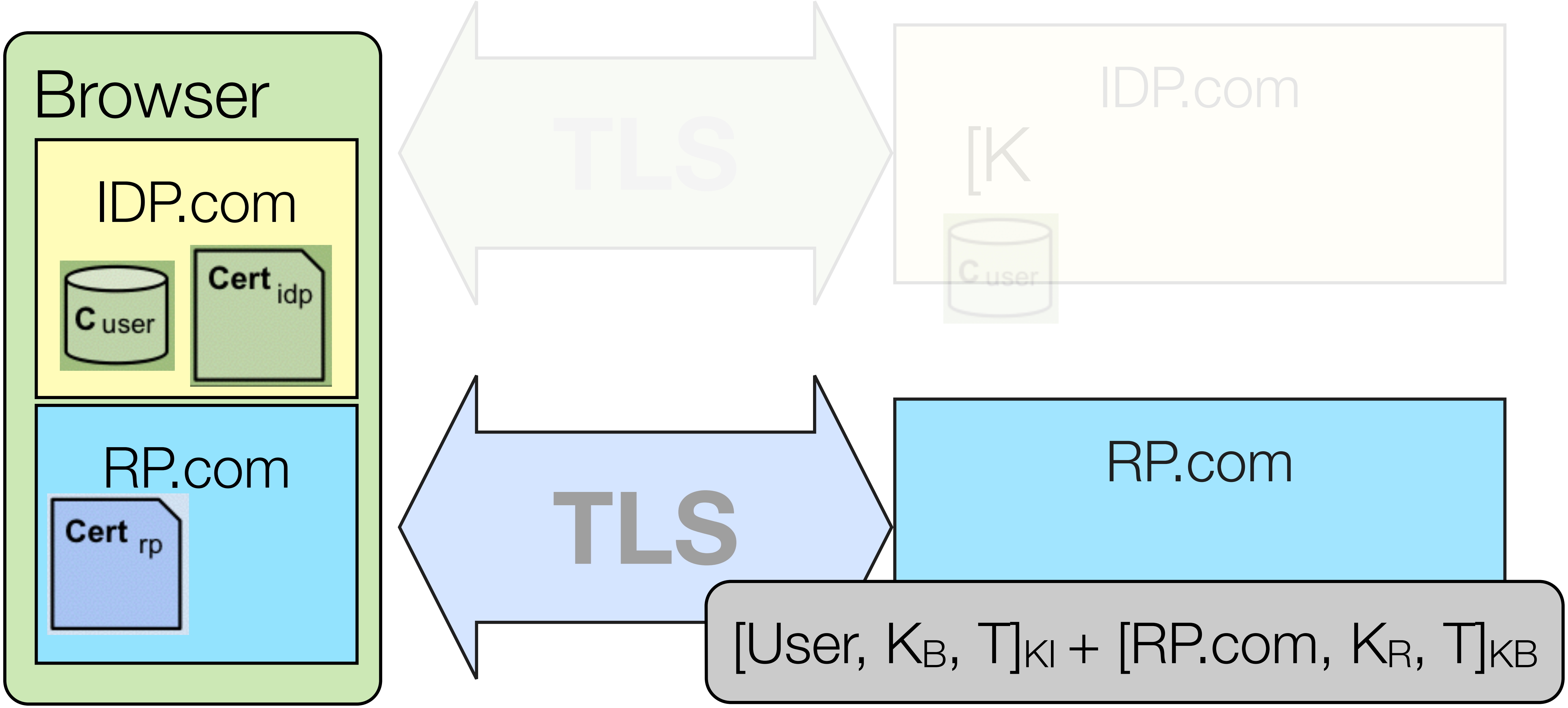
4. User wants to log into RP, Browser creates identity assertion



[User, K_B, T]$_{KI}$ + [RP.com, K_R, T]$_{KB}$

Persona-OBC-Local protocol

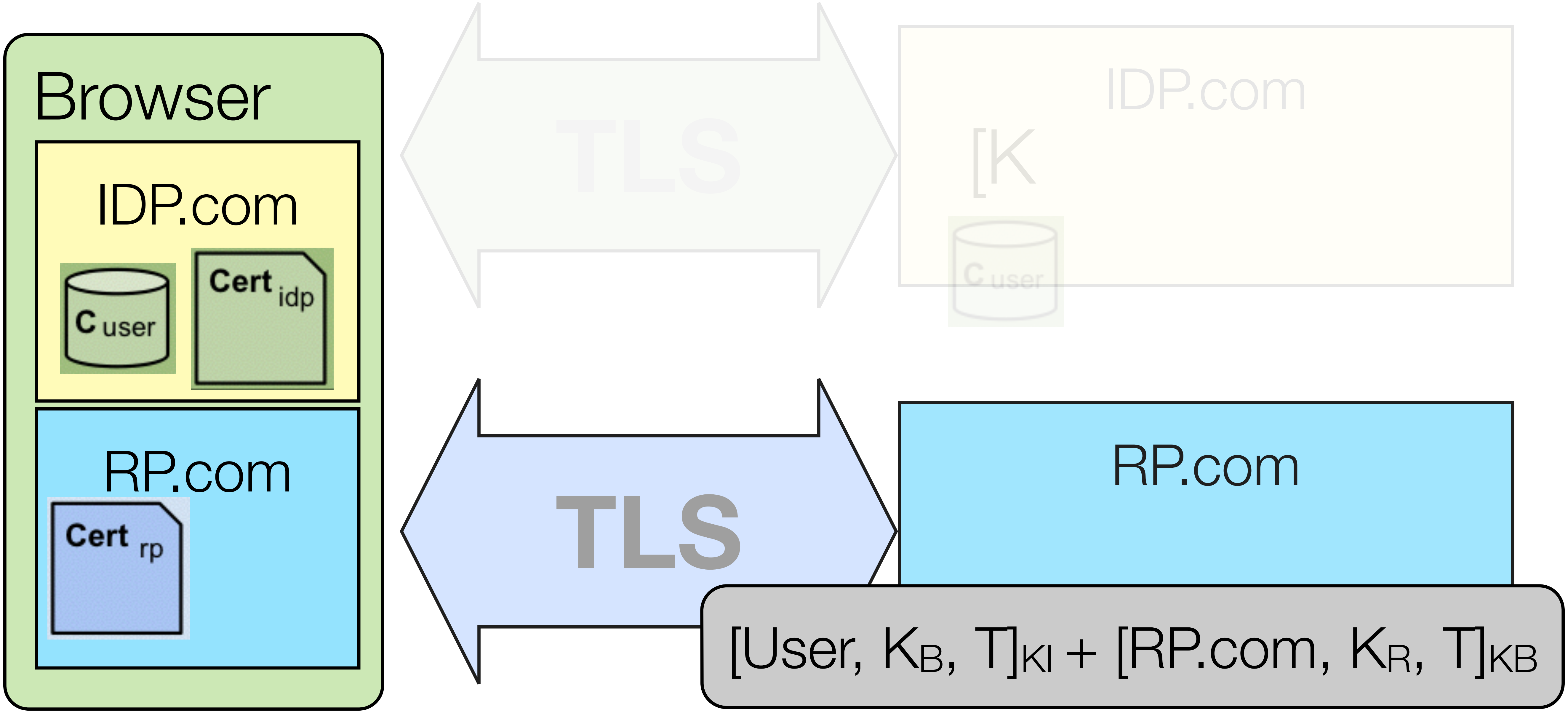
4. User wants to log into RP,
 Browser creates identity assertion



$[\text{User}, K_B, T]_{KI} + [\text{RP.com}, K_R, T]_{KB}$

Persona-OBC-Local protocol

5. RP verifies assertion



Browser

IDP.com

C_{user}

$Cert_{idp}$

RP.com

$Cert_{rp}$

TLS

IDP.com

[K

C_{user}

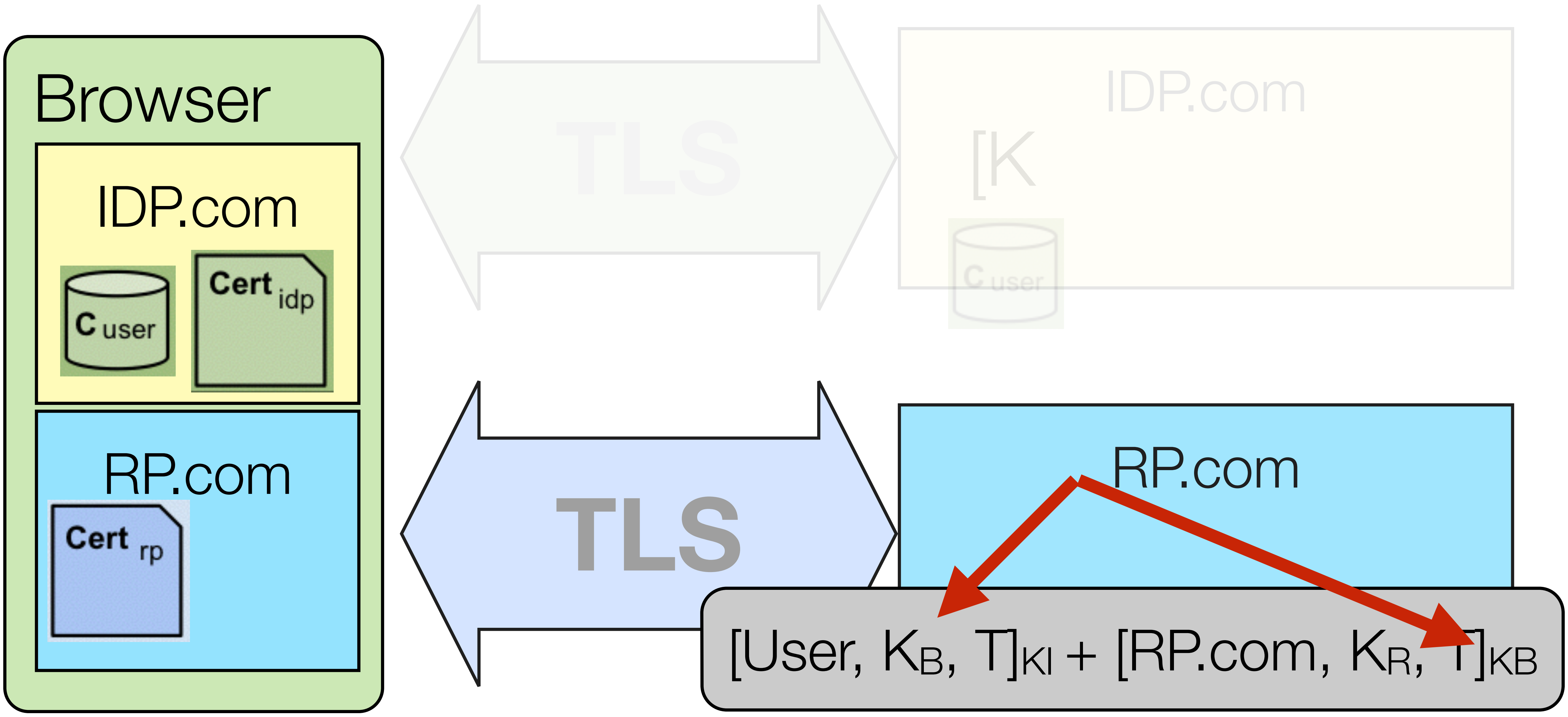
TLS

RP.com

$[User, K_B, T]_{KI} + [RP.com, K_R, T]_{KB}$

Persona-OBC-Local protocol

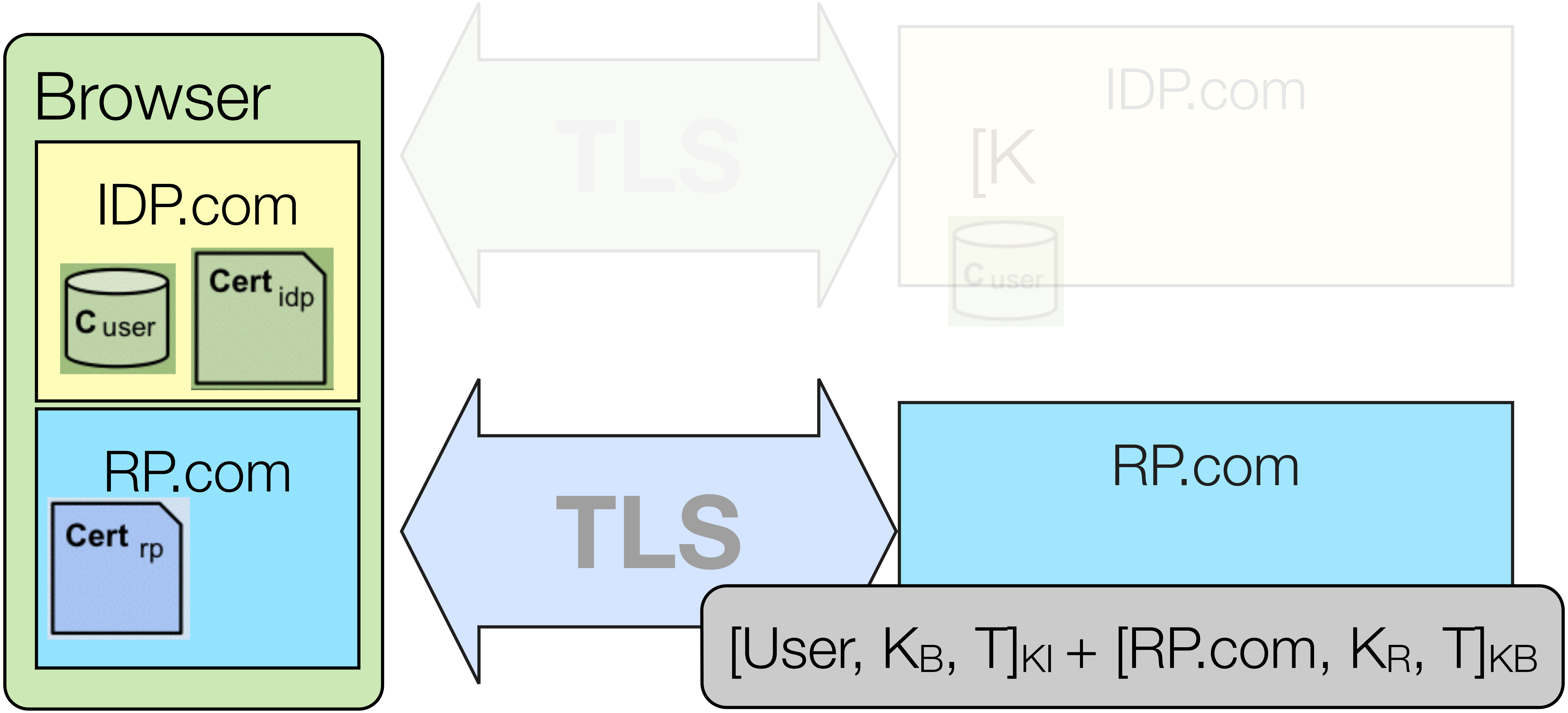
5. RP verifies assertion



$[\text{User}, K_B, T]_{KI} + [\text{RP.com}, K_R, T]_{KB}$

Persona-OBC-Local protocol

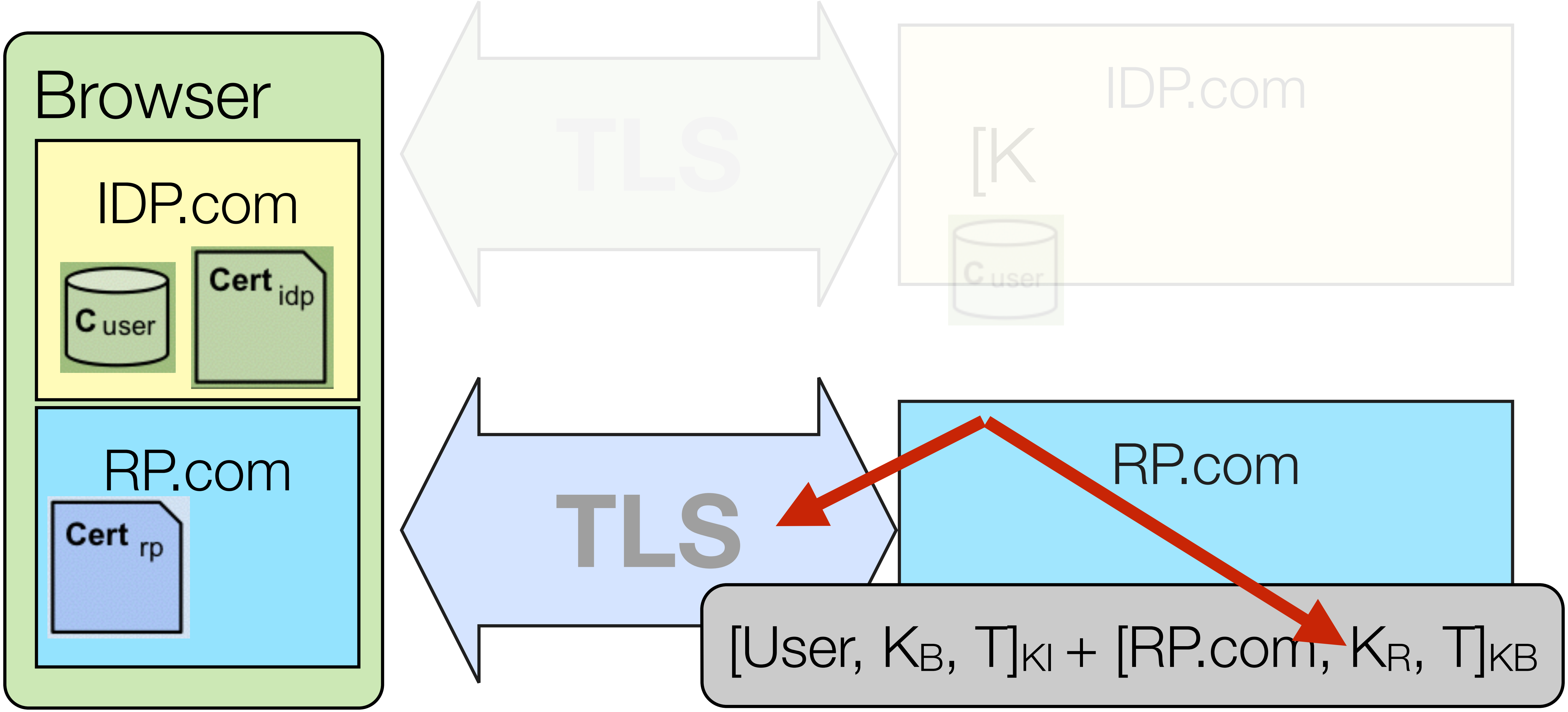
5. RP verifies assertion



$[\text{User}, K_B, T]_{KI} + [\text{RP.com}, K_R, T]_{KB}$

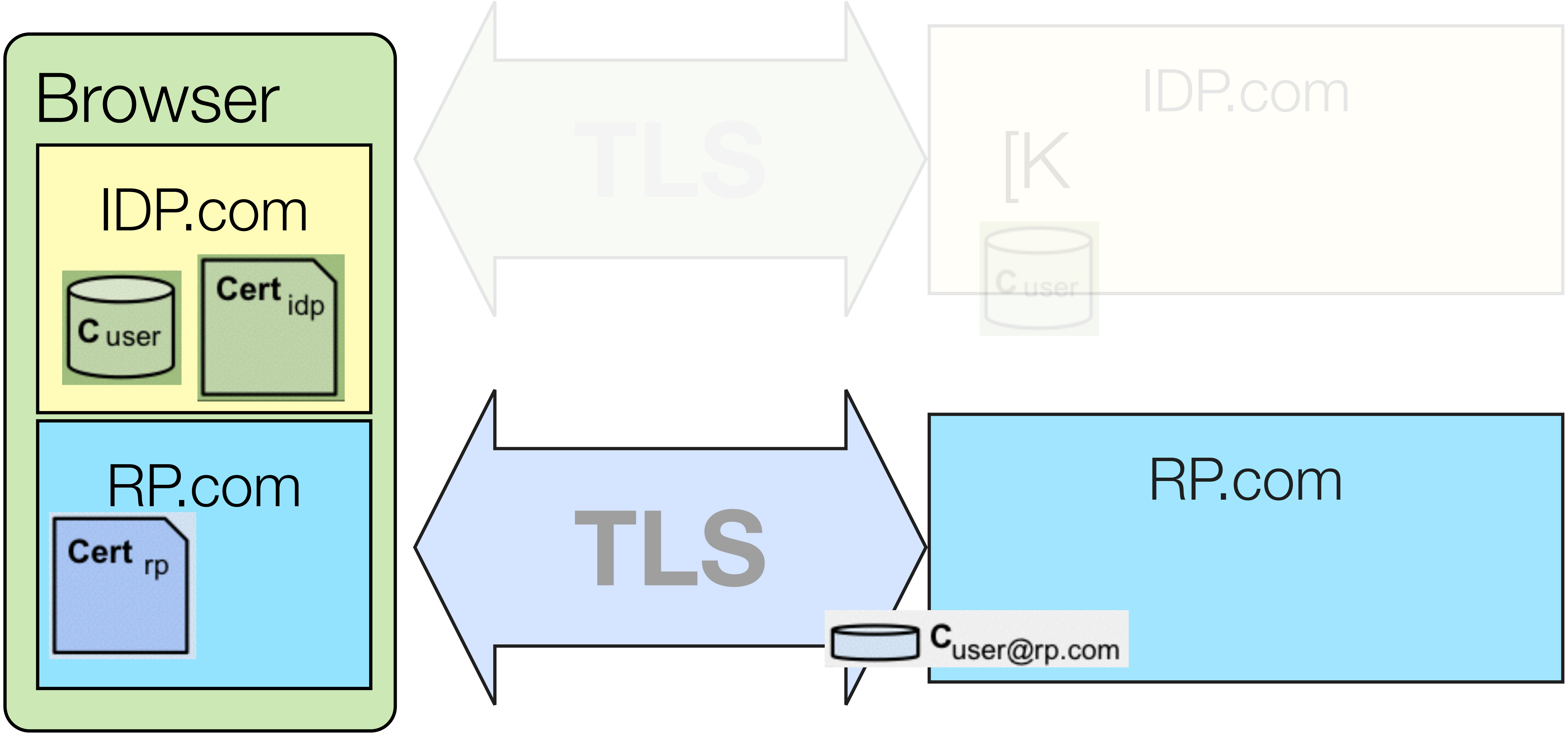
Persona-OBC-Local protocol

5. RP verifies assertion



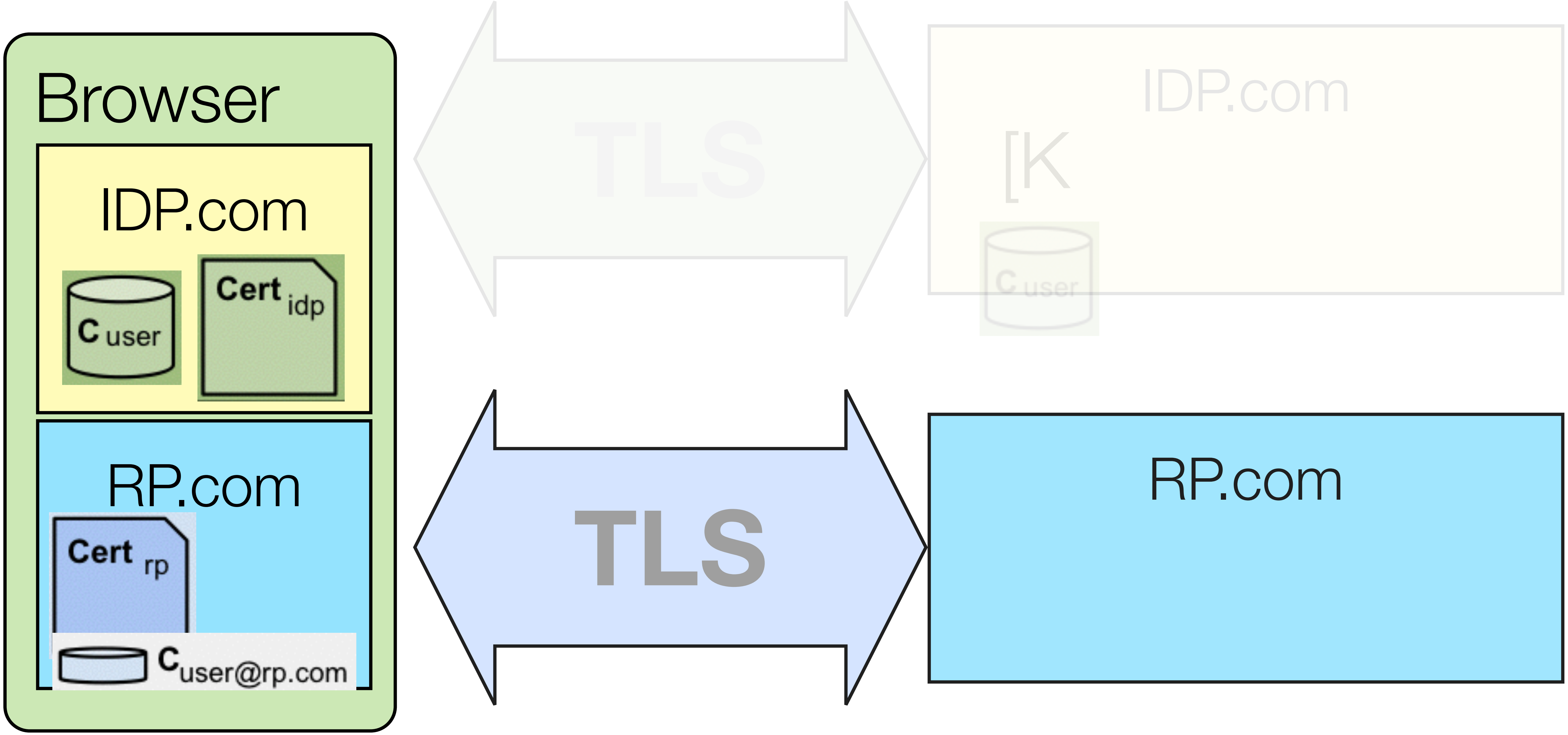
Persona-OBC-Local protocol

5. RP mints (channel-bound) cookie for user



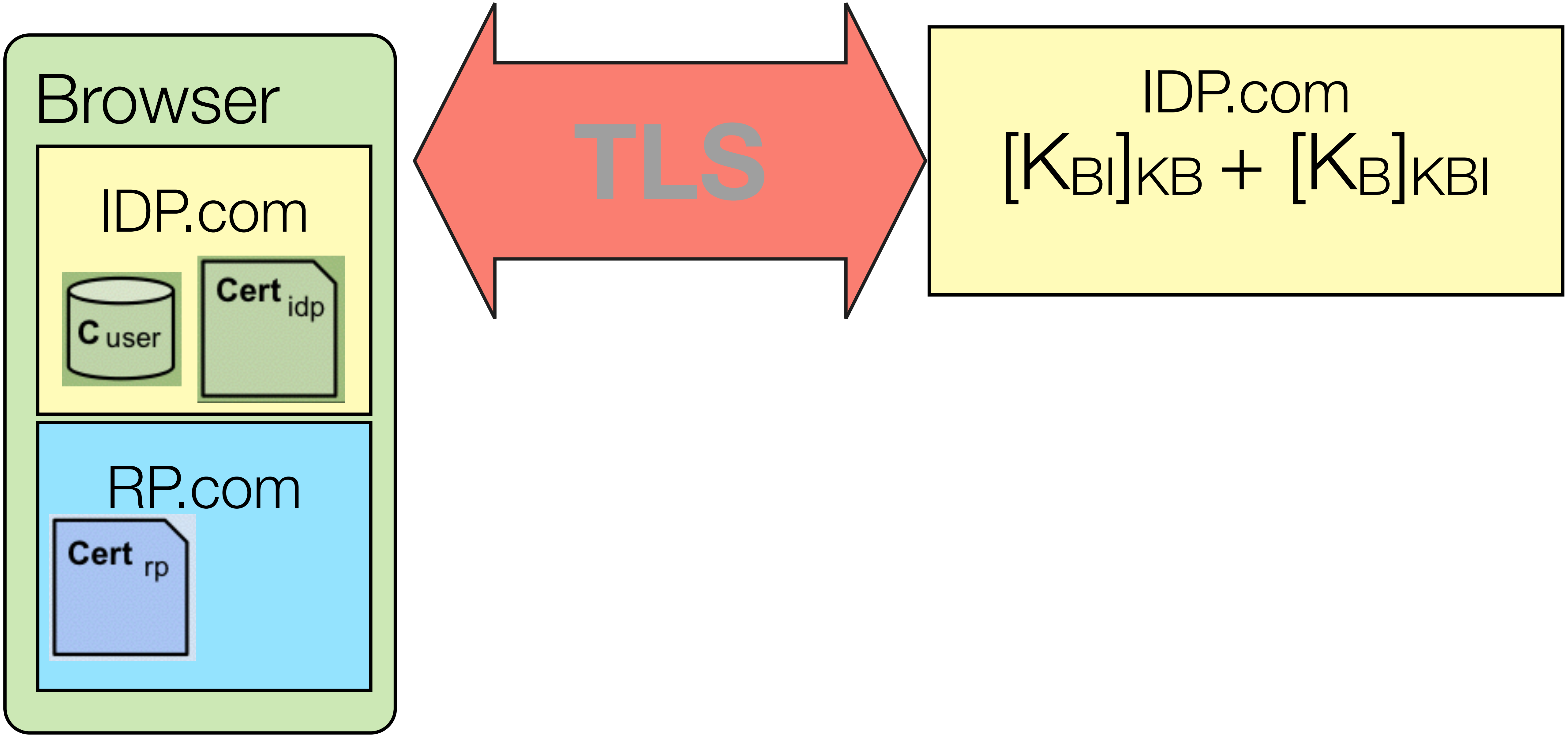
Persona-OBC-Local protocol

5. RP mints (channel-bound) cookie for user



Attack #1

- Attacker between browser and IDP



Browser

IDP.com

C_{user}

$Cert_{idp}$

RP.com

$Cert_{rp}$

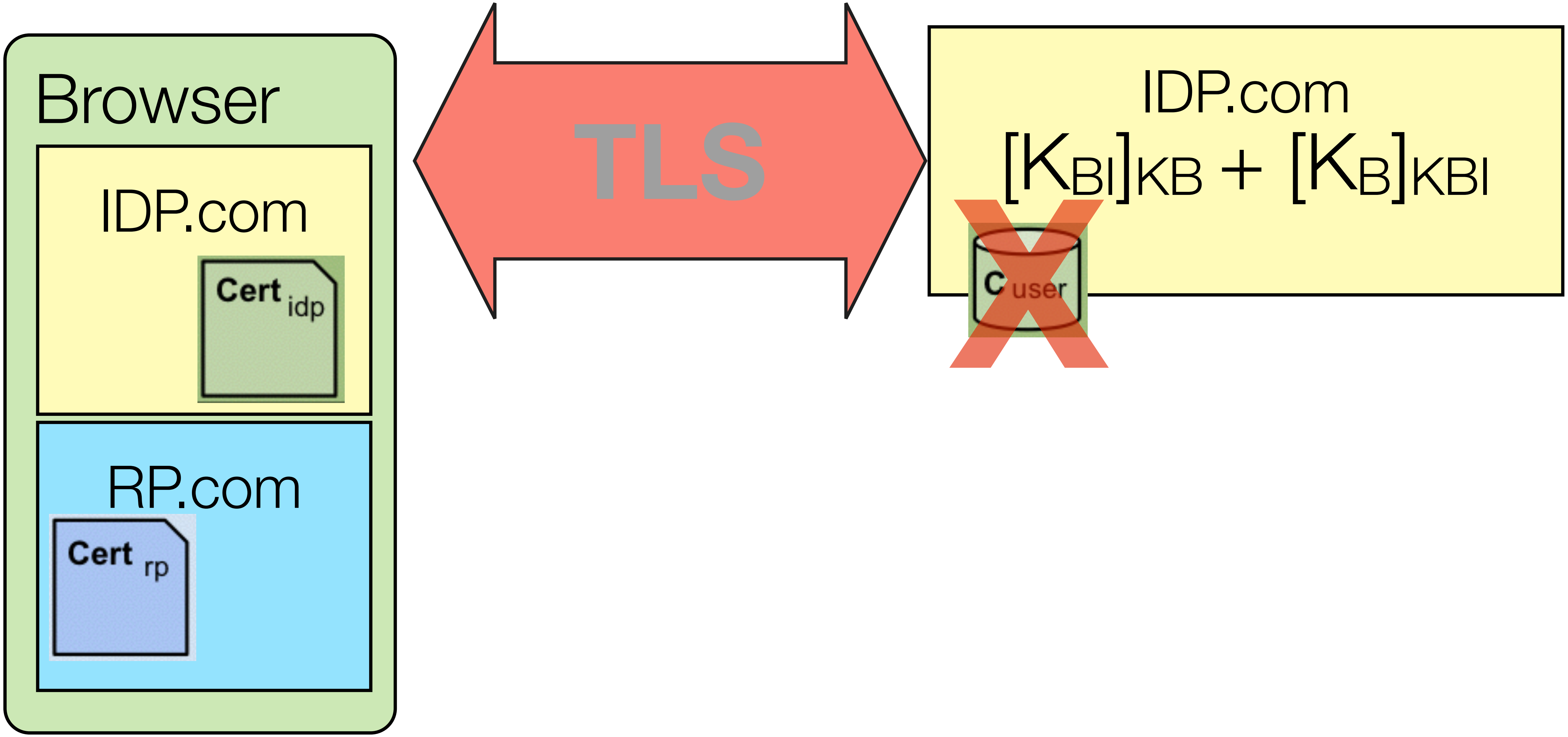
TLS

IDP.com

$[K_{BI}]_{KB} + [K_{B}]_{KBI}$

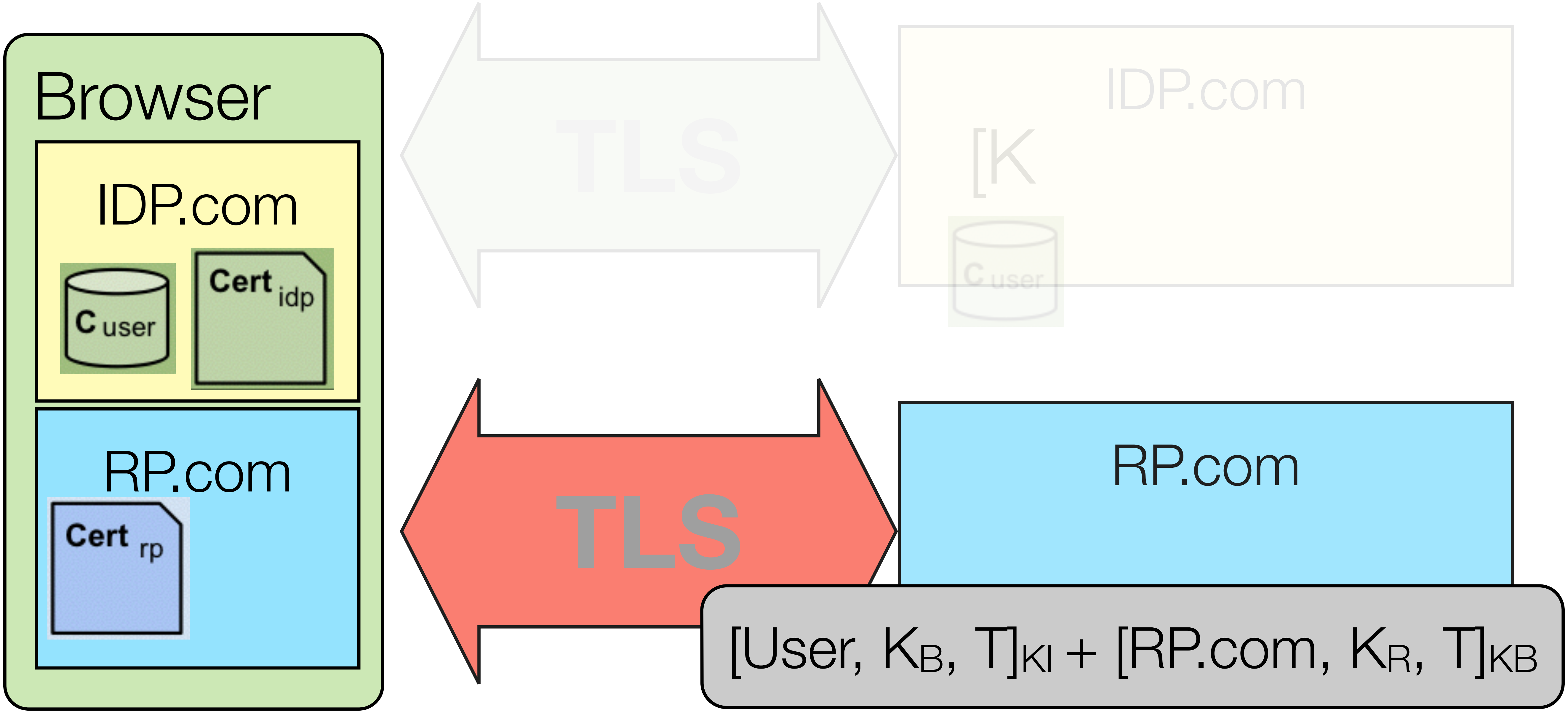
Attack #1

- Attacker between browser and IDP



Attack #2

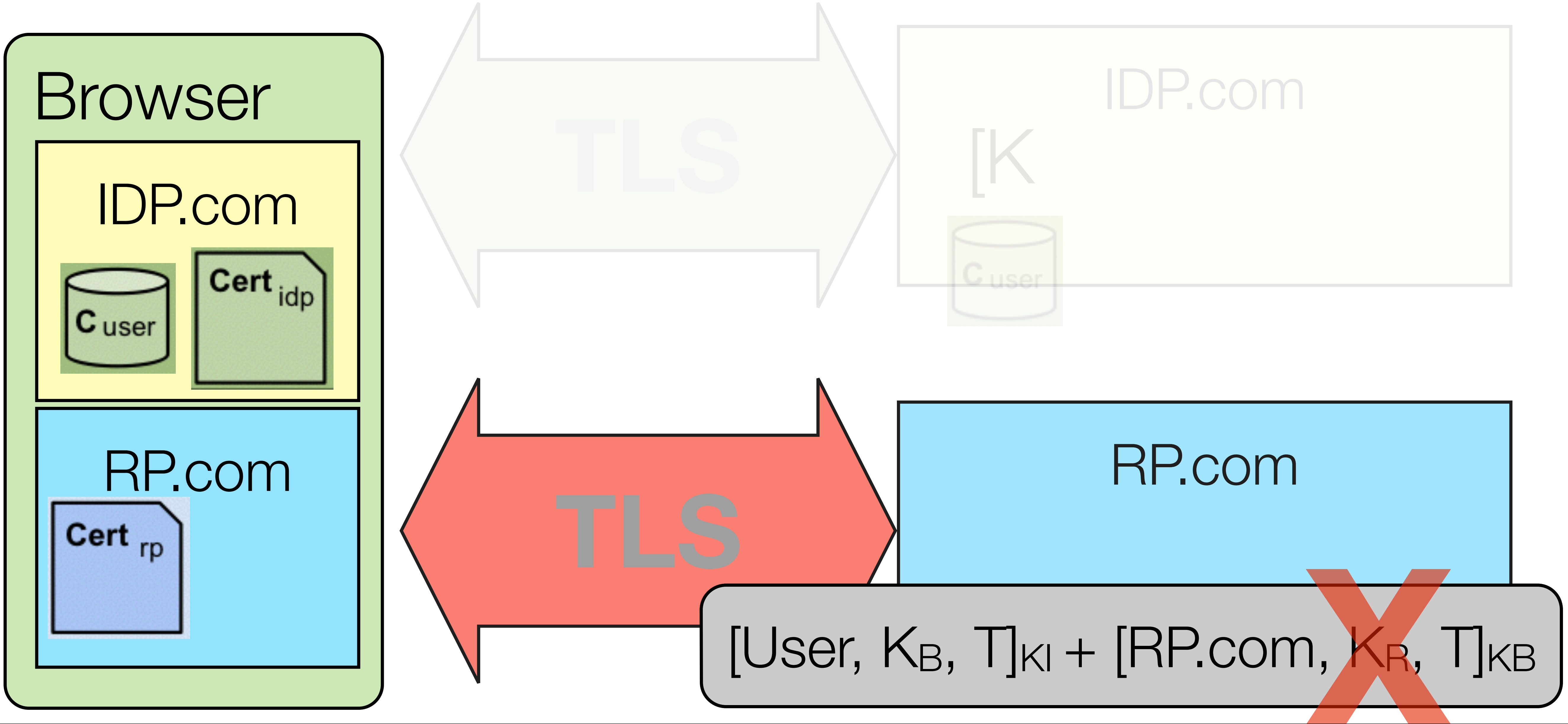
- Attacker between browser and RP



[User, K_B, T]$_{KI}$ + [RP.com, K_R, T]$_{KB}$

Attack #2

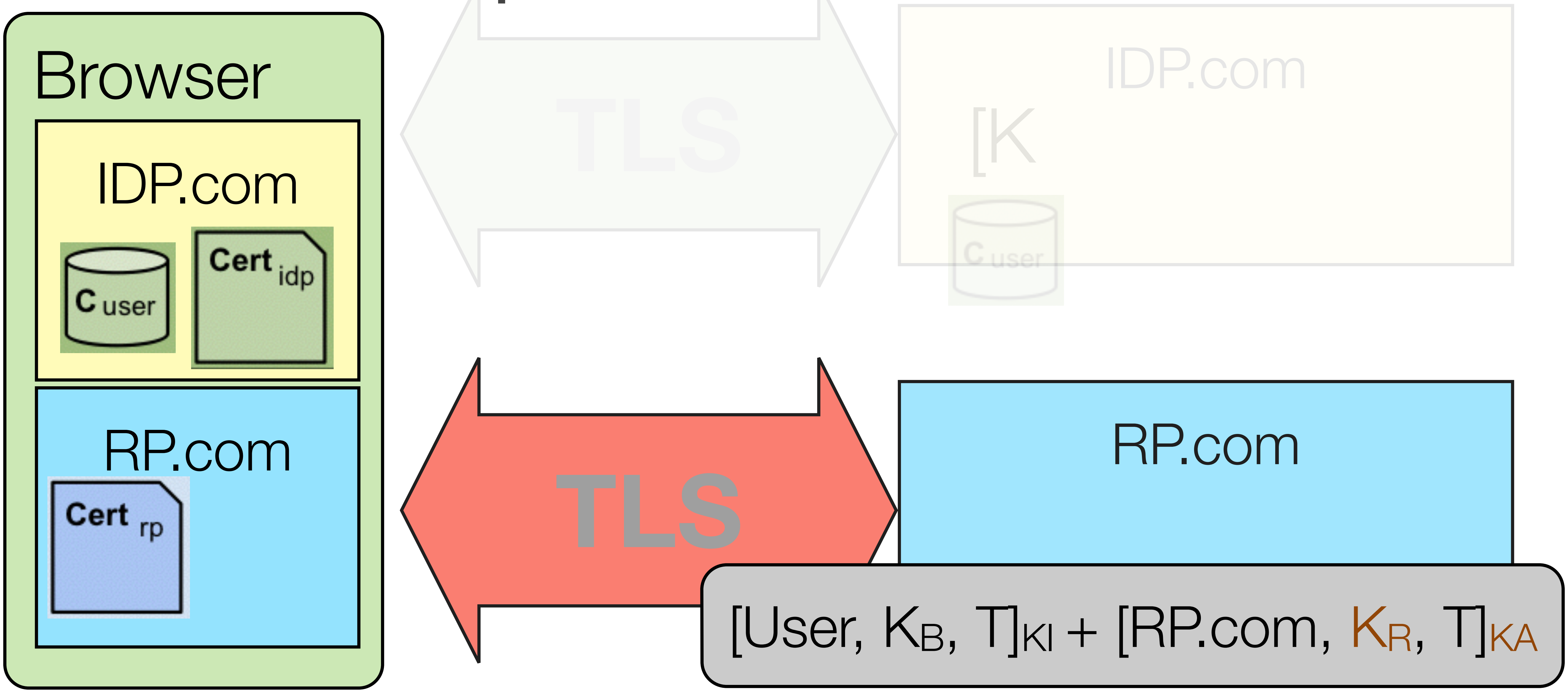
- Attacker between browser and RP



$[\text{User}, K_B, T]_{KI} + [\text{RP.com}, \cancel{K_R}, T]_{KB}$

Attack #3

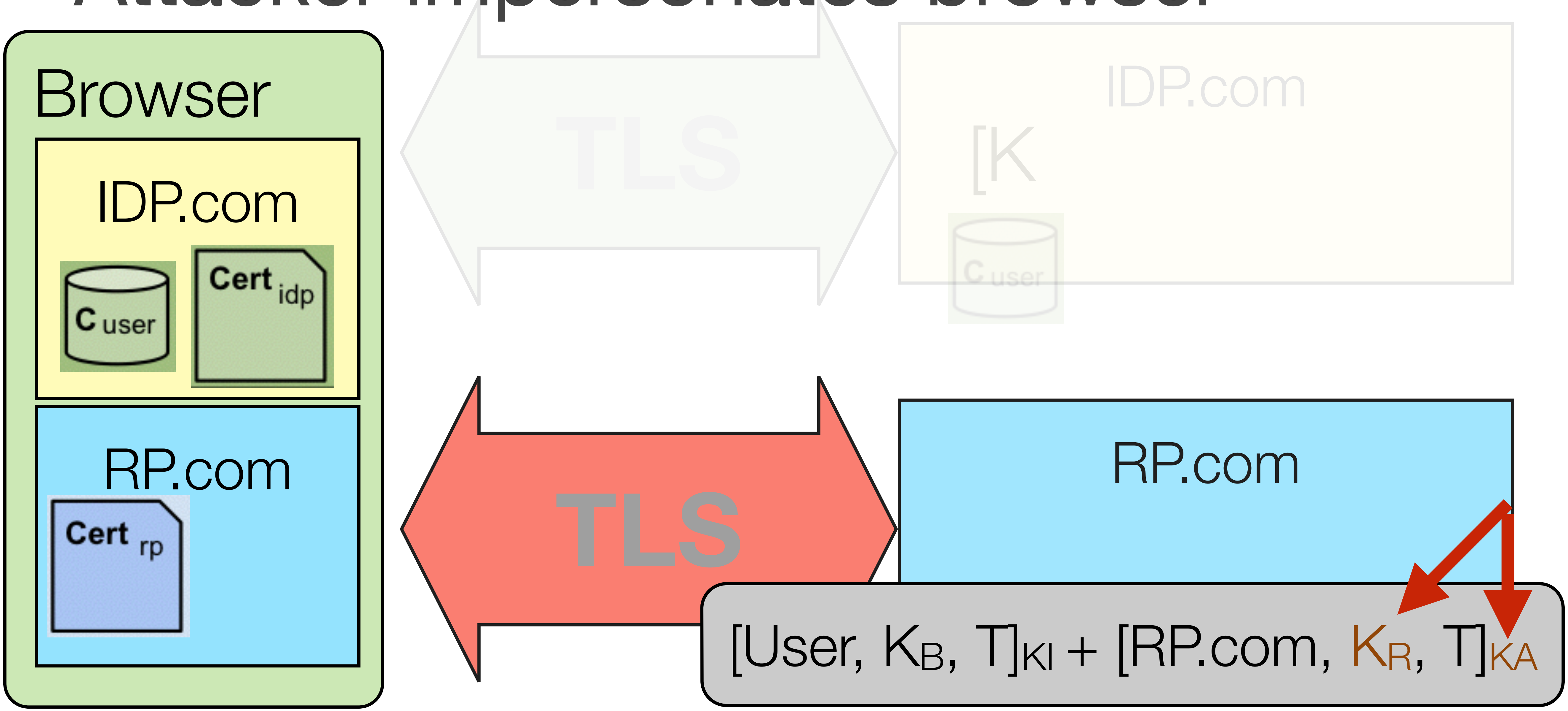
- Attacker between browser and RP
- Attacker impersonates browser



$[\text{User}, K_B, T]_{KI} + [\text{RP.com}, K_R, T]_{KA}$

Attack #3

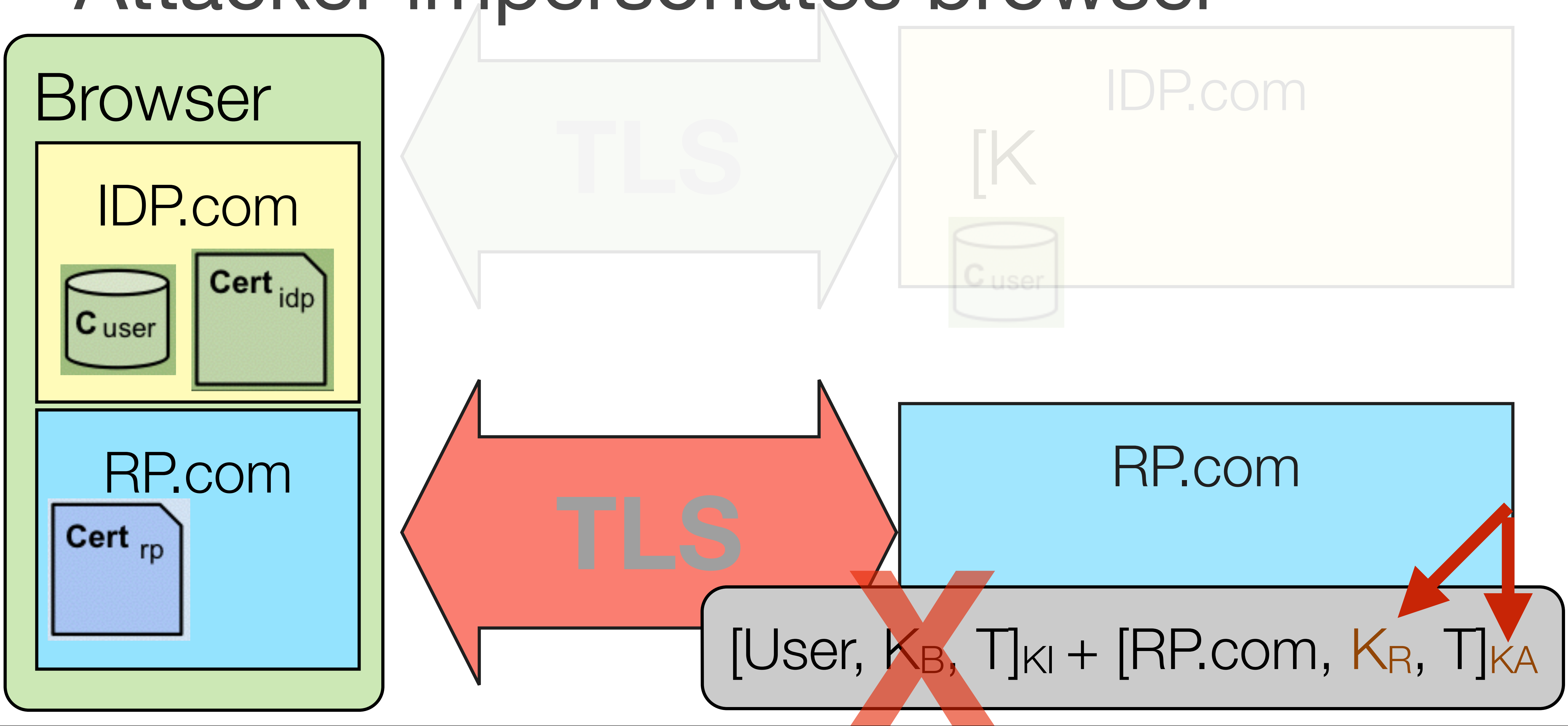
- Attacker between browser and RP
- Attacker impersonates browser



[User, K_B, T]$_{KI}$ + [RP.com, K_R, T]$_{KA}$

Attack #3

- Attacker between browser and RP
- Attacker impersonates browser



Protocol implementation

- Proof of concept IDP and RP implementations for Persona-OBC-Local
- Both written in Python
  - Use Nexus Authorization Logic proof checker to verify assertions
- BAN logic formalization of both protocols
  - Local and Central variants

Conclusion

- Two persona extensions
  - Better MITM protection for identity assertions
  - Leverage channel between IDP and user to create channel between user and RP
  - RP uses a different key than IDP to communicate with the user (for privacy)

Questions?

mdietz@gmail.com