Encrypted Access Logging for Online Accounts: Device Attributions without Device Tracking

Carolina Ortega Pérez*, Alaa Daffalla*, and Thomas Ristenpart

*Equal contribution by both authors.

Cornell University | CORNELL TECH

Accounts

Importance of Account Security Interfaces (ASIs)

Importance of Account Security Interfaces (ASIs)

Account compromise is a frequent problem for survivors

- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

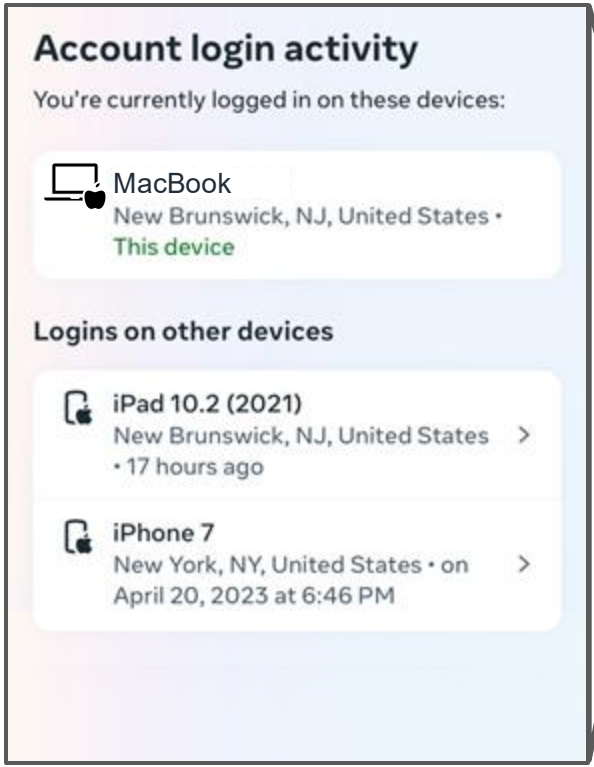
Importance of Account Security Interfaces (ASIs)

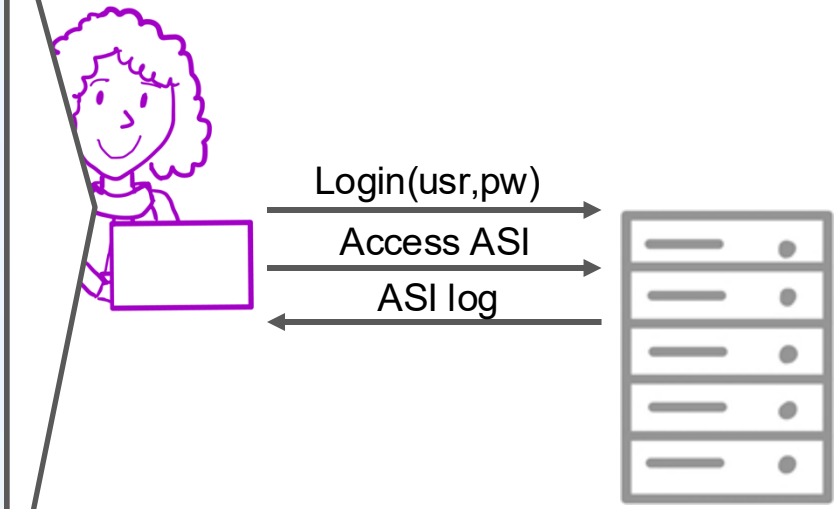
CETA
CLINIC TO END TECH ABUSE

Account compromise is a frequent problem for survivors

- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

Importance of Account Security Interfaces (ASIs)



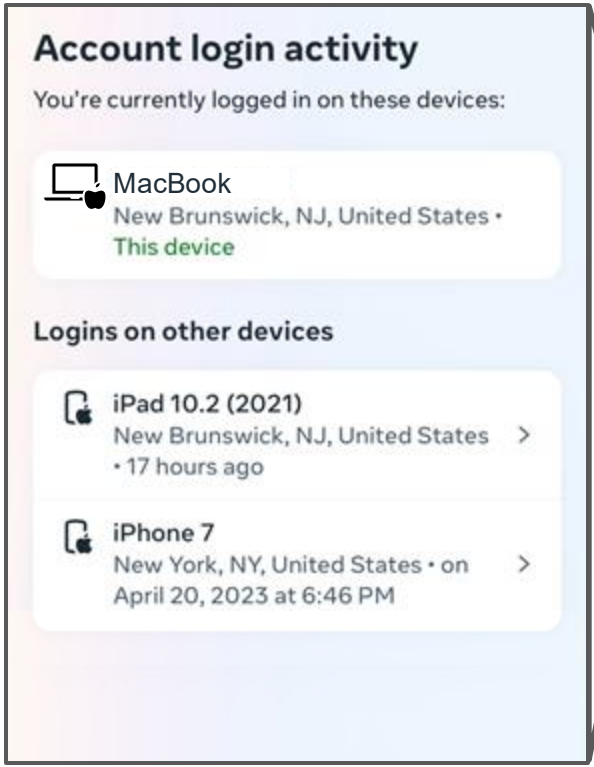


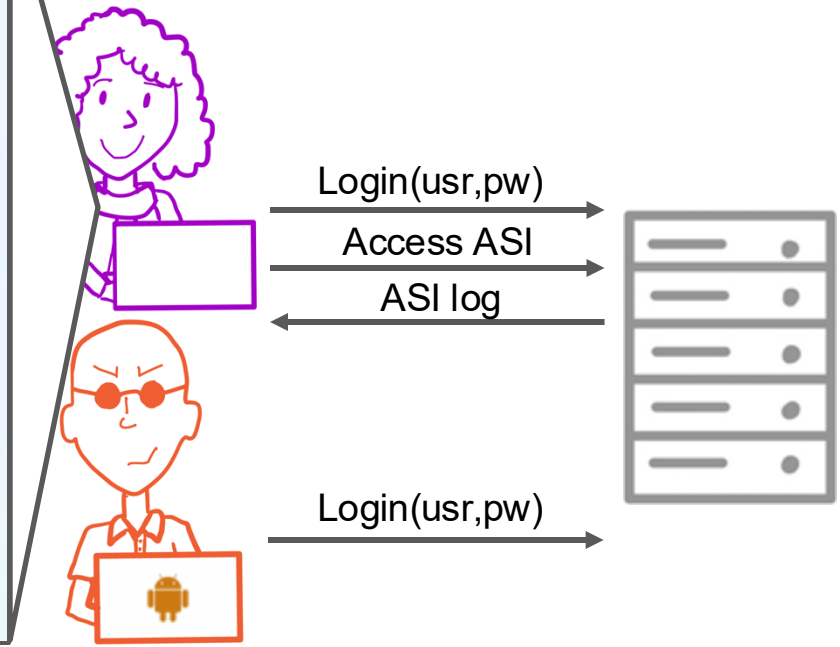
CETA
CLINIC TO END TECH ABUSE

Account compromise is a frequent problem for survivors

- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

Importance of Account Security Interfaces (ASIs)



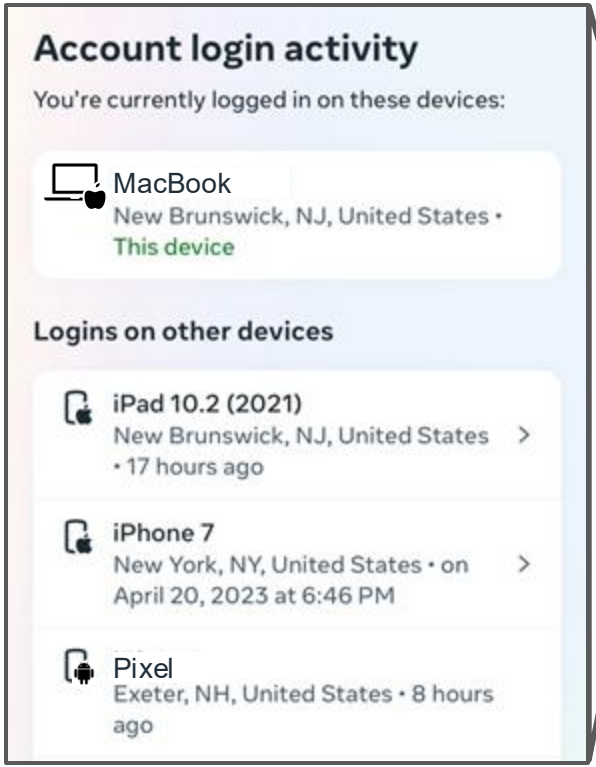


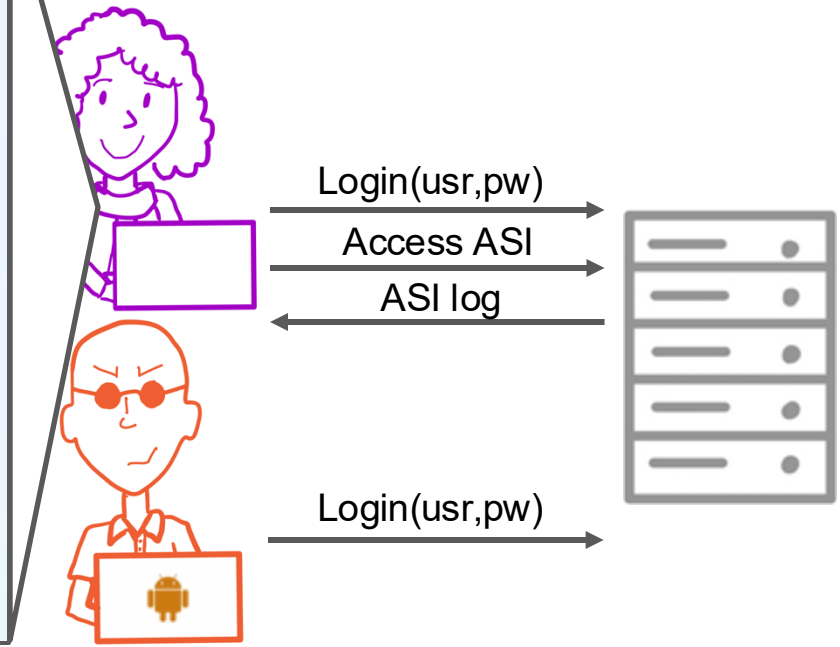
CETA
CLINIC TO END TECH ABUSE

Account compromise is a frequent problem for survivors

- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

Importance of Account Security Interfaces (ASIs)



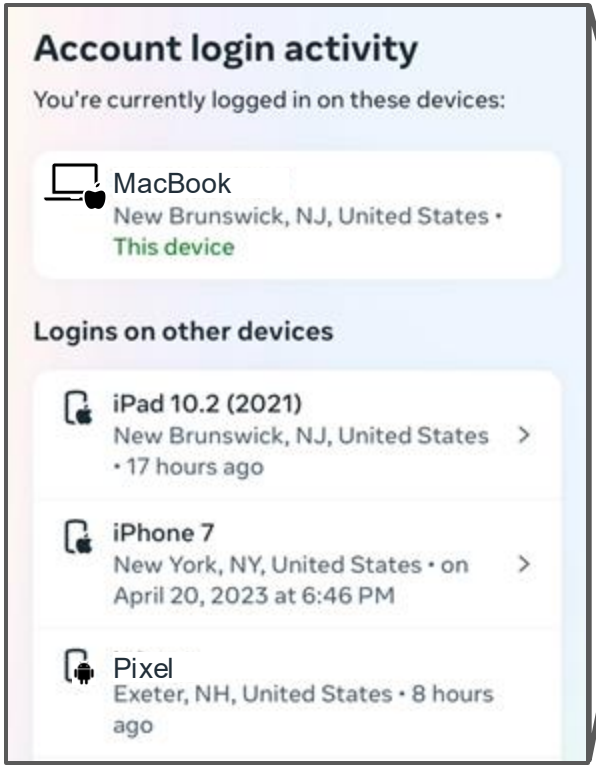


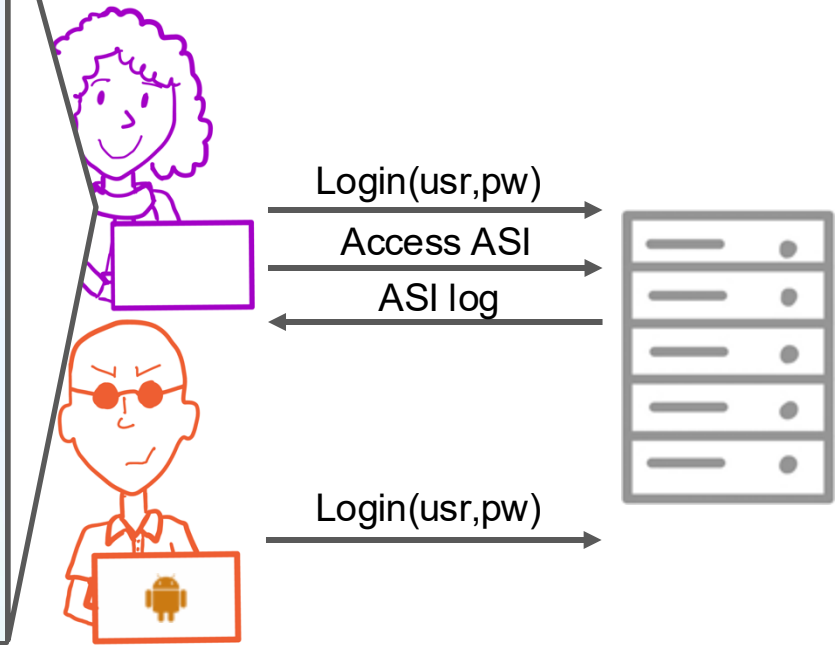
CETA
CLINIC TO END TECH ABUSE

Account compromise is a frequent problem for survivors

- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

Importance of Account Security Interfaces (ASIs)





CETA
CLINIC TO END TECH ABUSE

Account compromise is a frequent problem for survivors

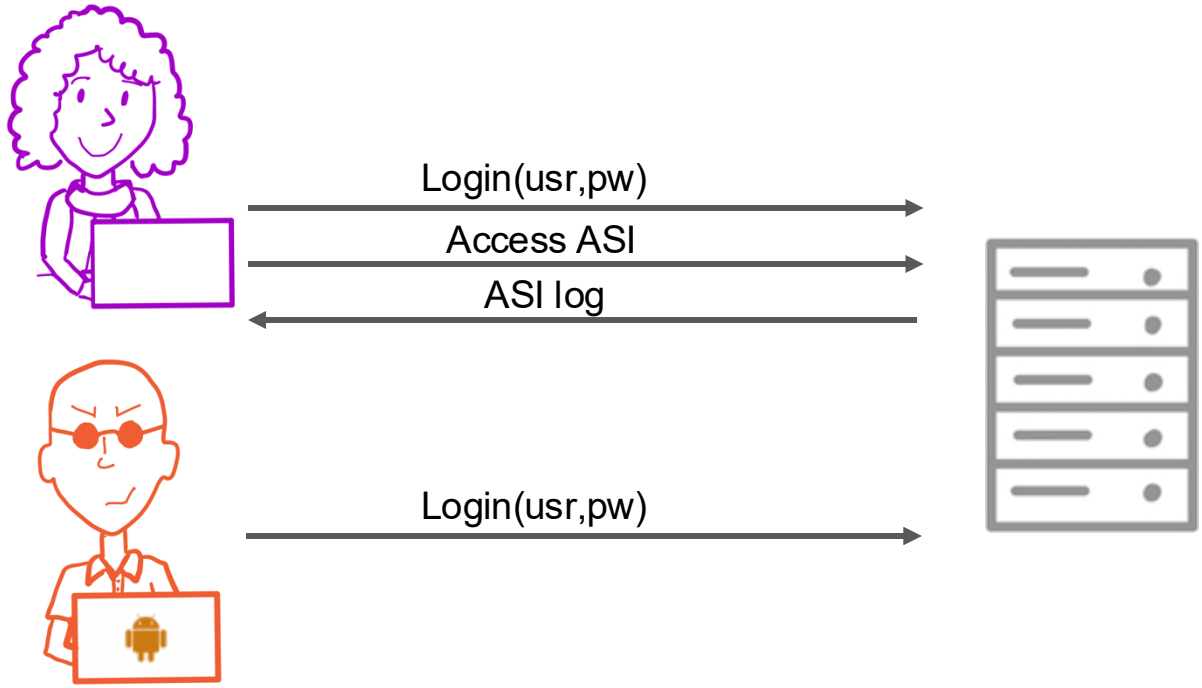
- Abusers can often bypass authentication
- May not be safe to remove abuser
- Survivors often need to know whether access has occurred

ASIs critically important for user safety

ASIs are easy to spoof

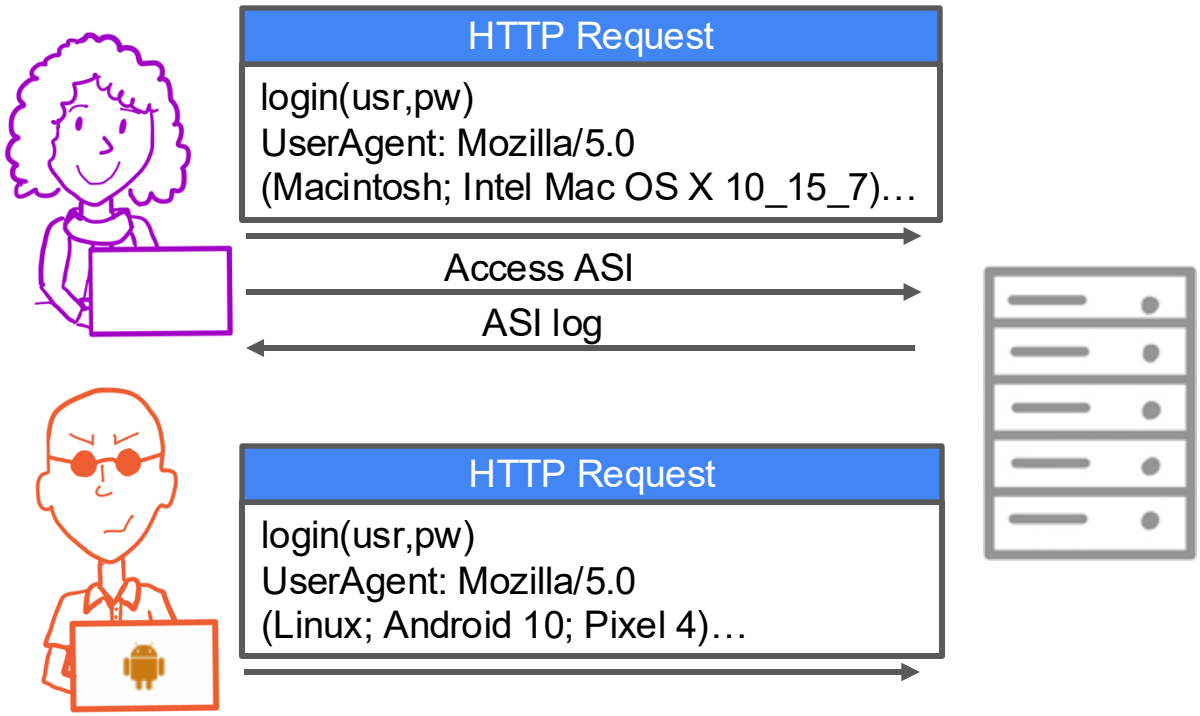
Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs are easy to spoof



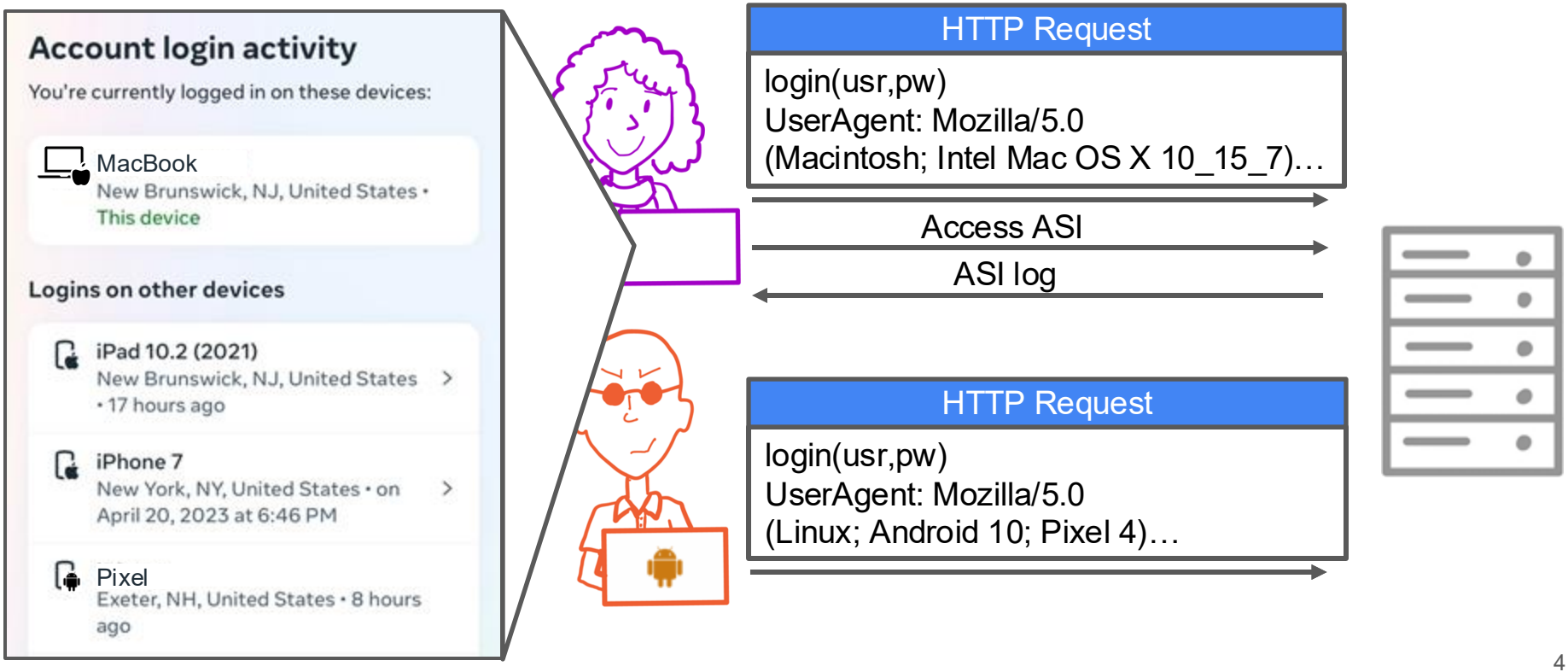
Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs are easy to spoof



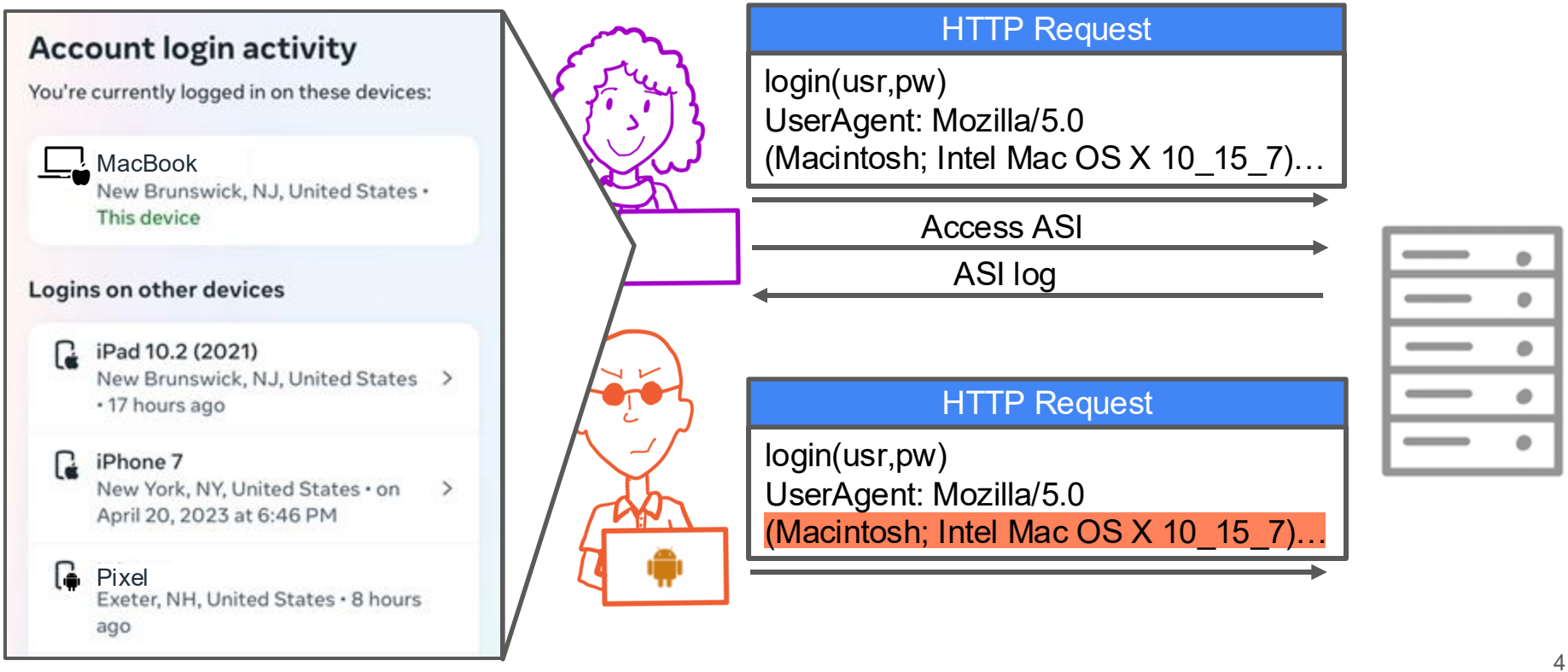
Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs are easy to spoof



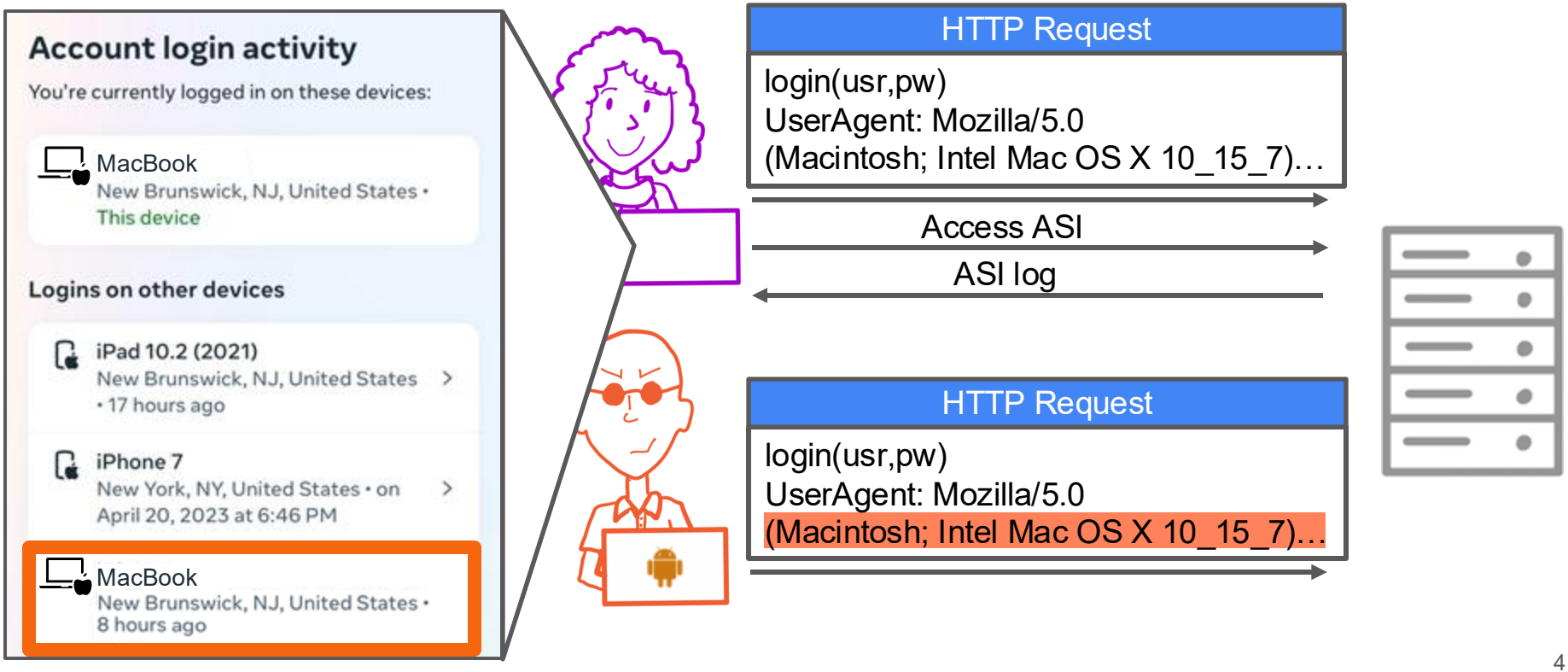
Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs are easy to spoof



Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs are easy to spoof



Account Security Interfaces: Important, Unintuitive, and Untrustworthy. Daffalla et al (2023)

ASIs integrity is a problem…

…what can we do about it?

Our work: Towards providing ASI integrity

We propose **client-side-encrypted access logging (CSAL)** protocols

- Agnostic to authentication mechanisms
- Do not rely on external services
- Require OS support (browsers are untrusted)

Our work: Towards providing ASI integrity

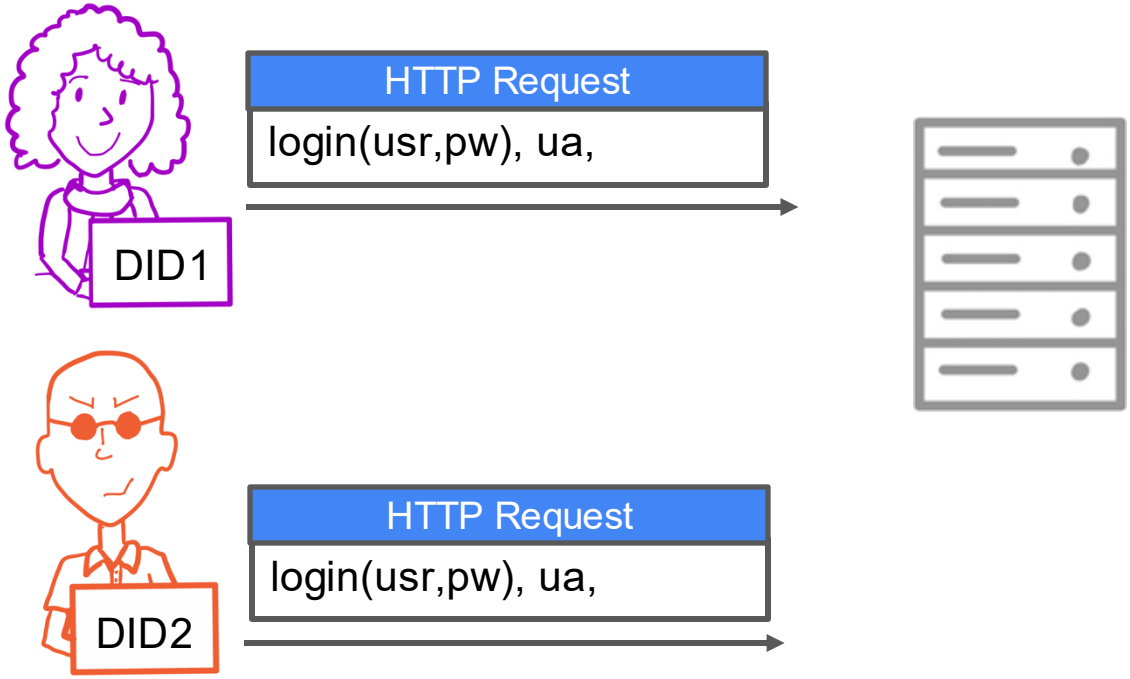
We propose **client-side-encrypted access logging (CSAL)** protocols

- Agnostic to authentication mechanisms
- Do not rely on external services
- Require OS support (browsers are untrusted)

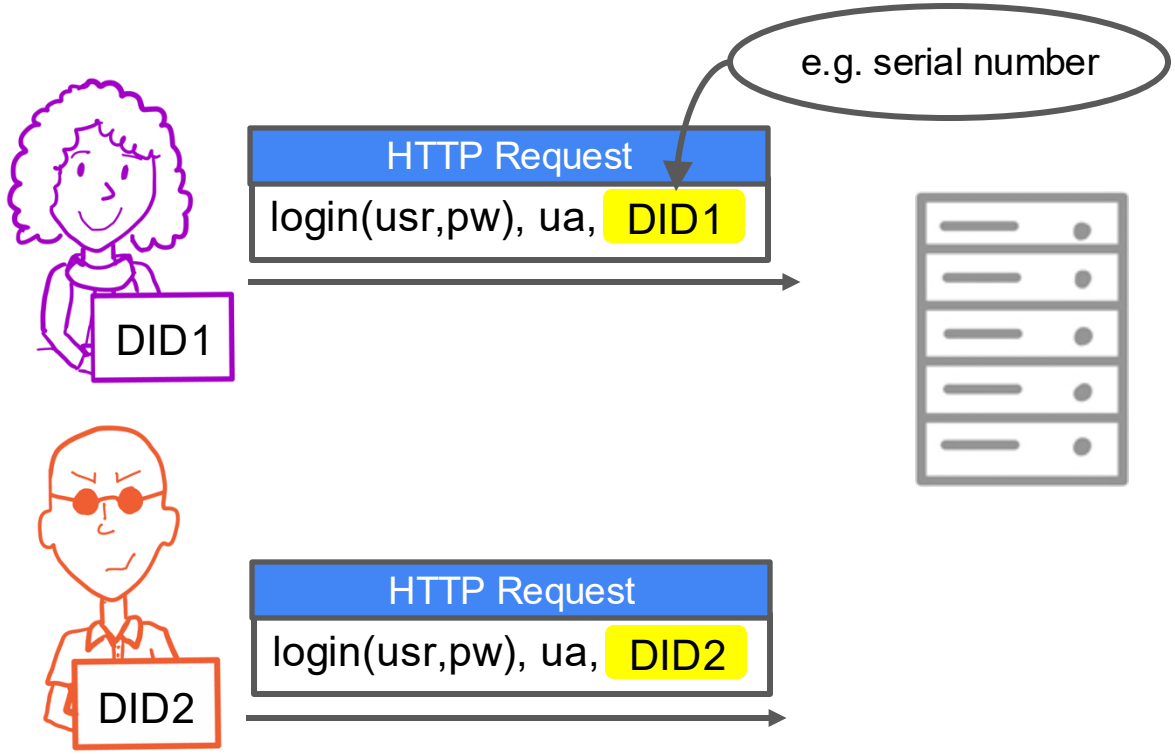
In the paper we:

- Formalize CSAL protocols
- Detail a full CSAL protocol and formally analyze it
- Proof-of-concept CSAL implementation and deployment considerations

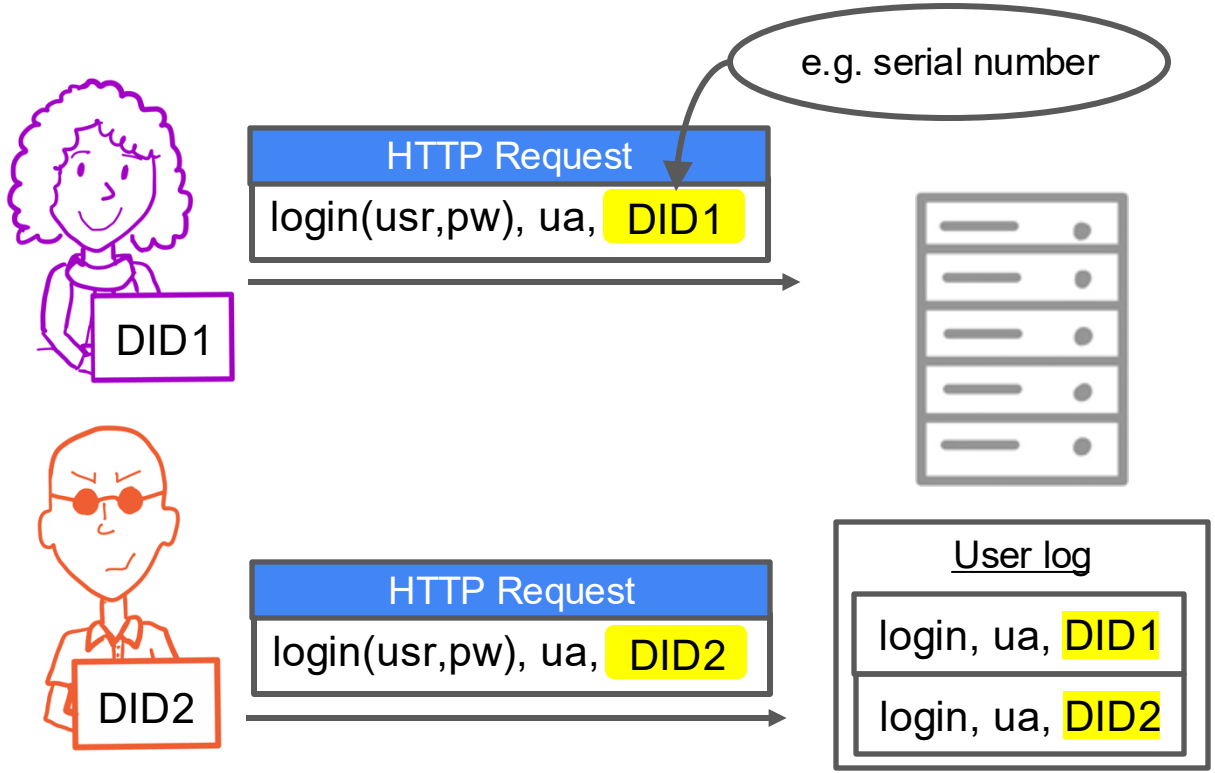
Straw approach: Adding reliable OS-provided information



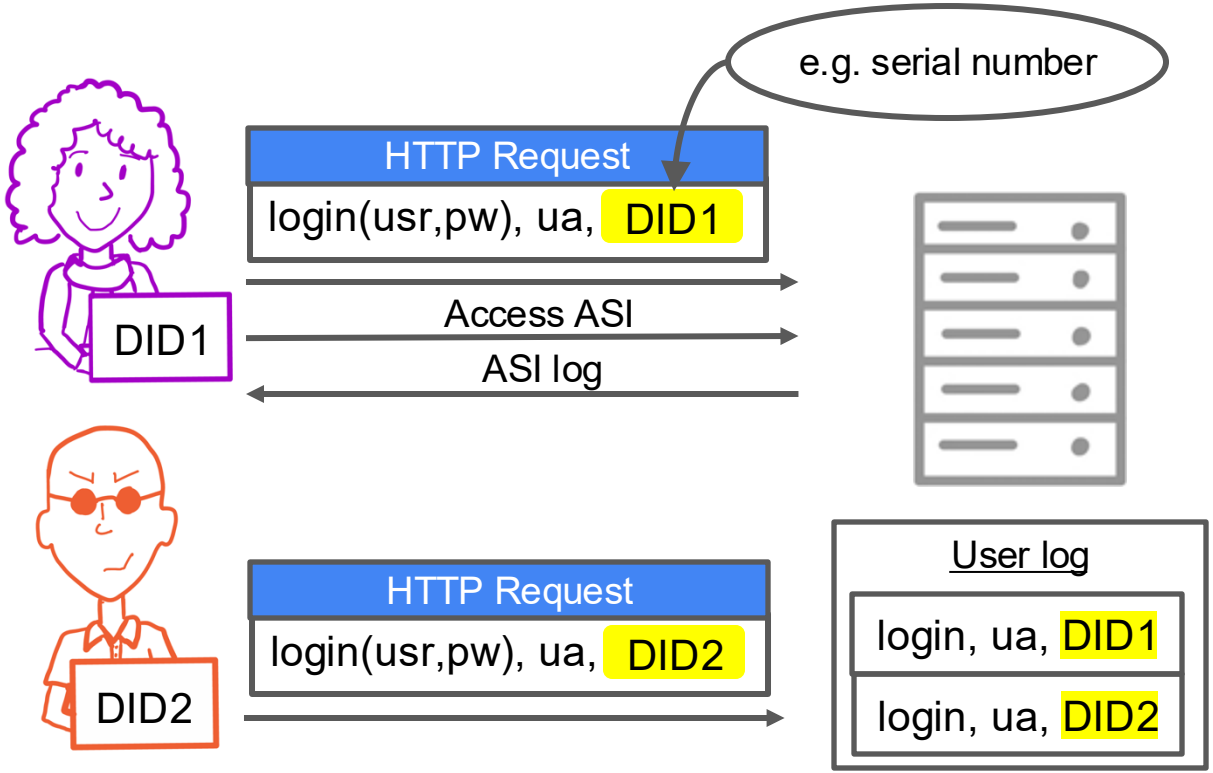
Straw approach: Adding reliable OS-provided information



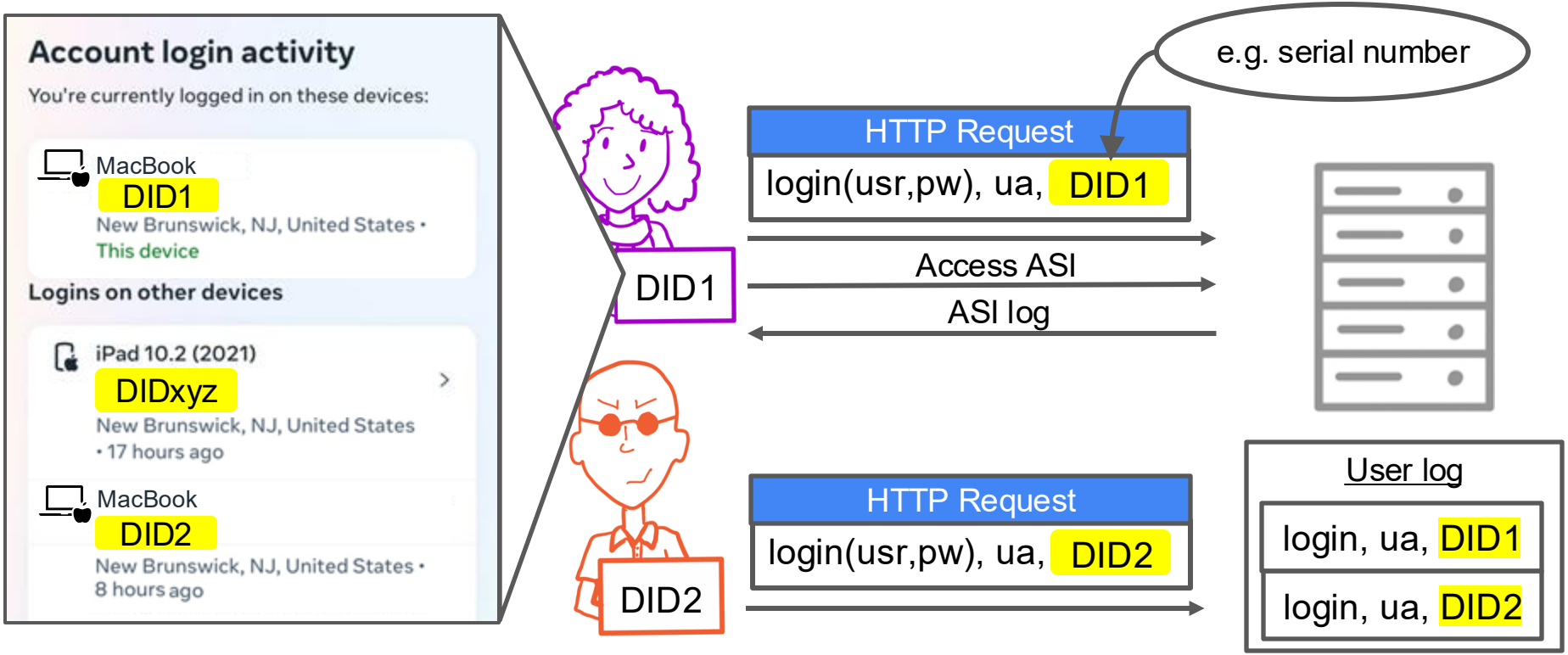
Straw approach: Adding reliable OS-provided information



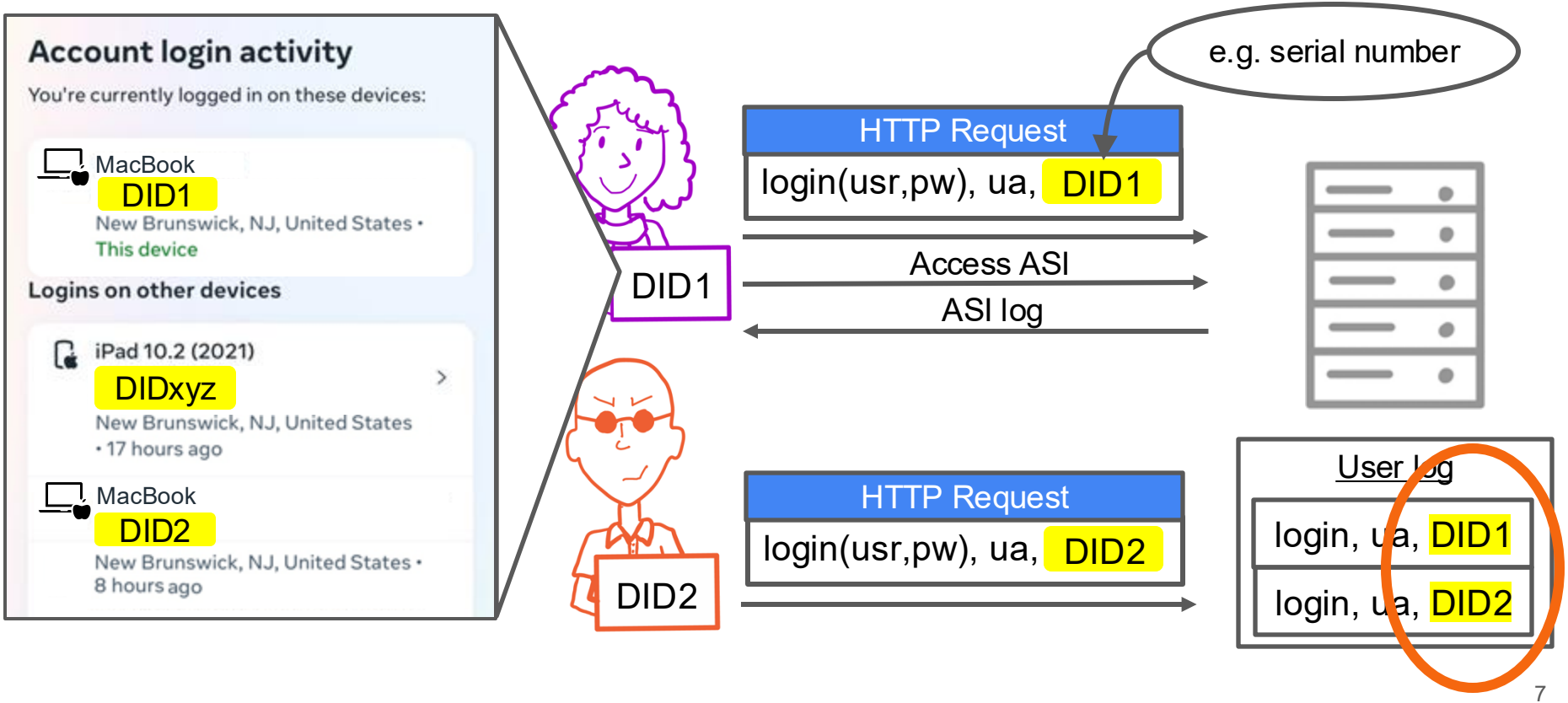
Straw approach: Adding reliable OS-provided information



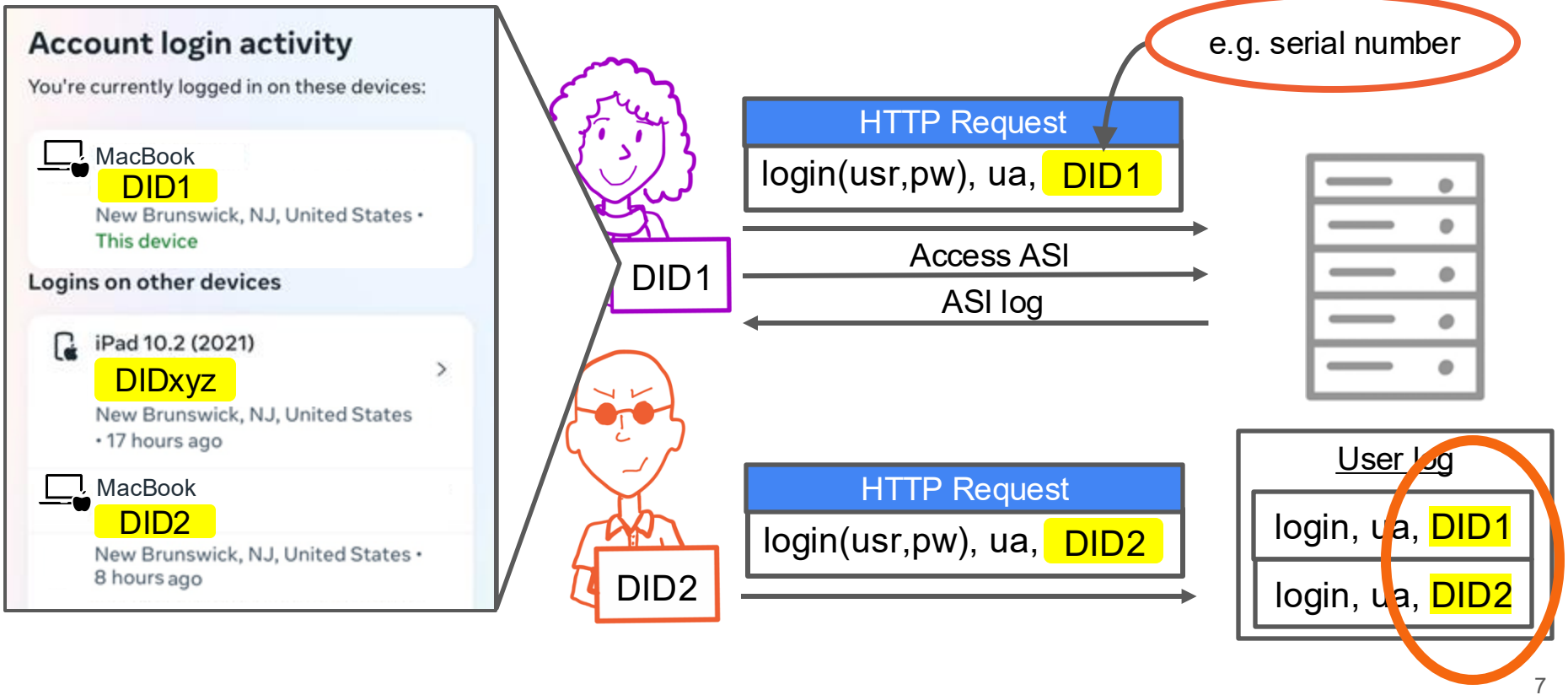
Straw approach: Adding reliable OS-provided information



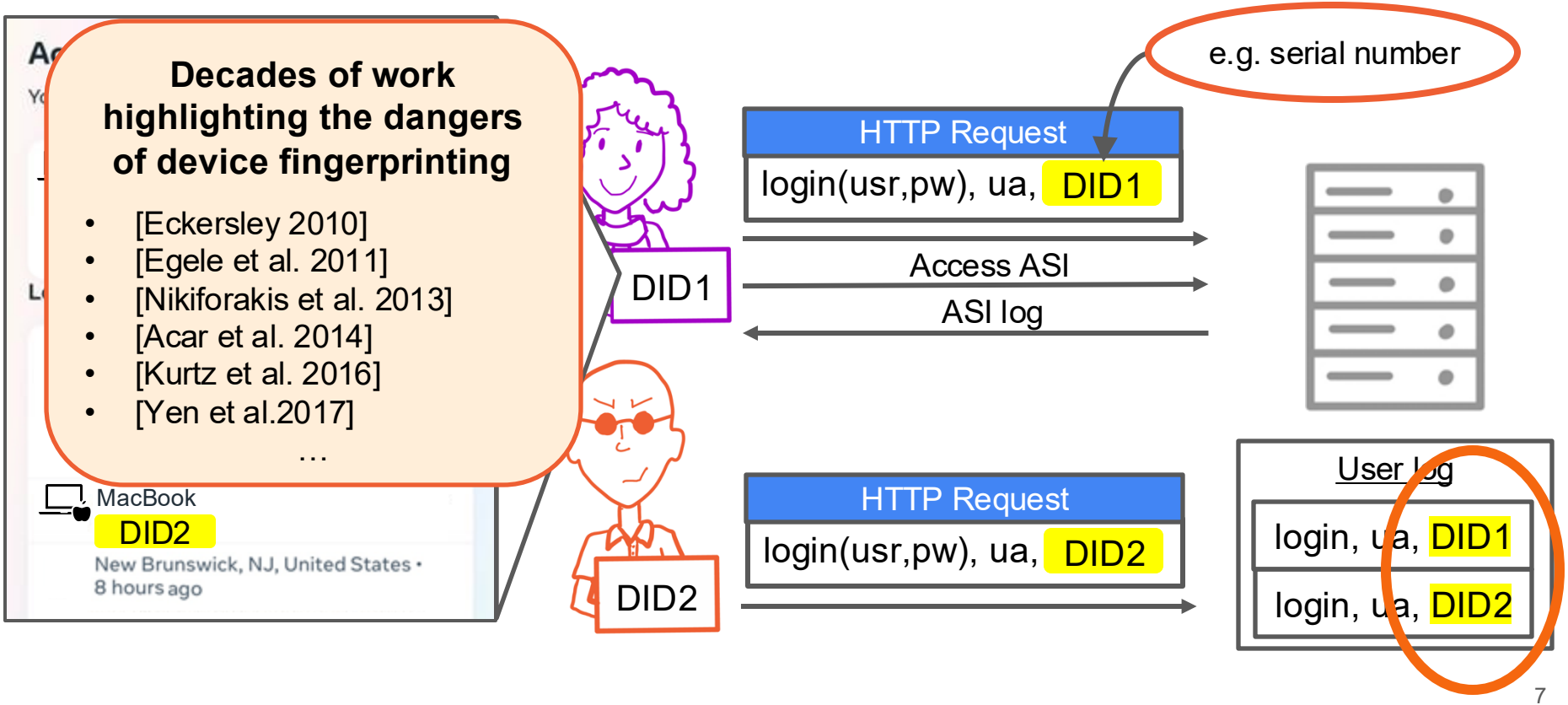
Straw approach: Adding reliable OS-provided information



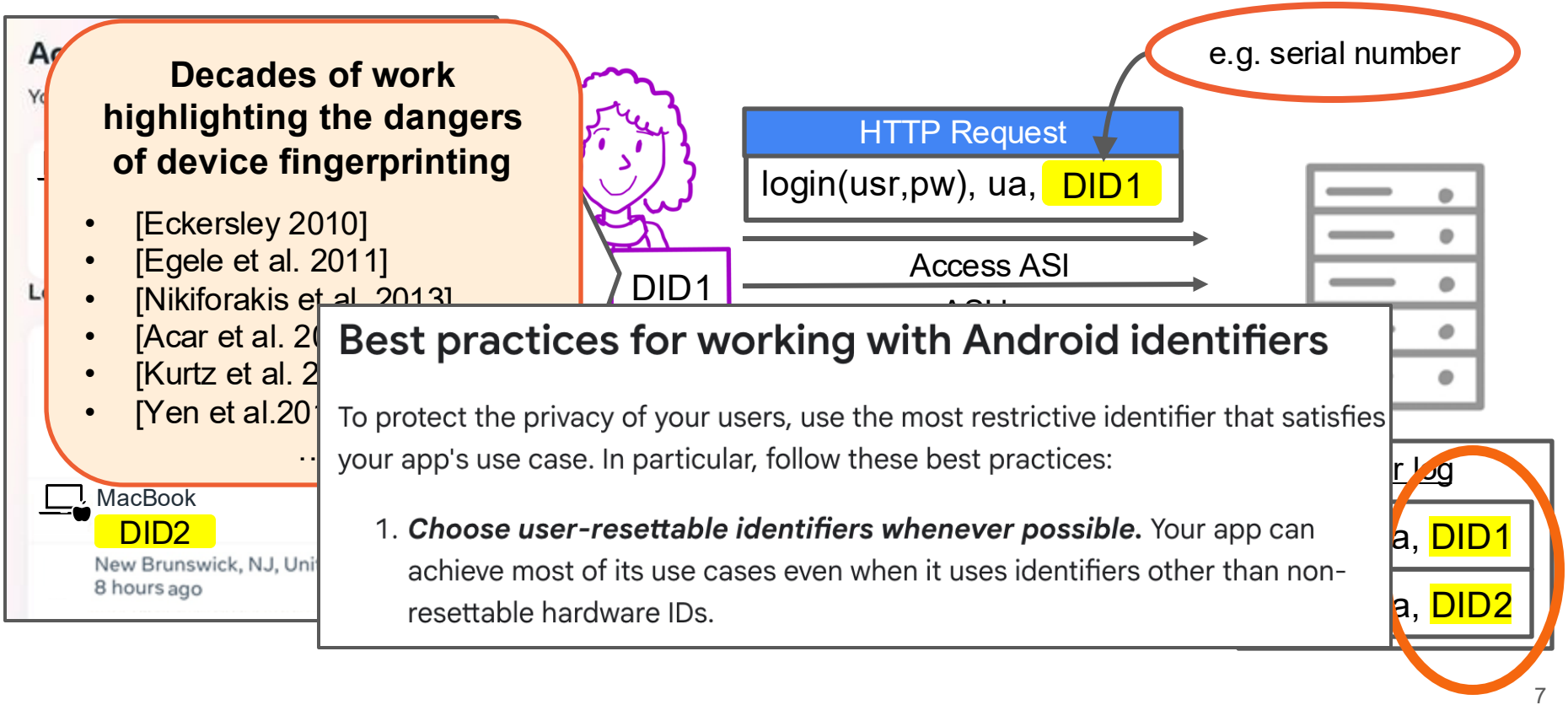
Straw approach: Adding reliable OS-provided information



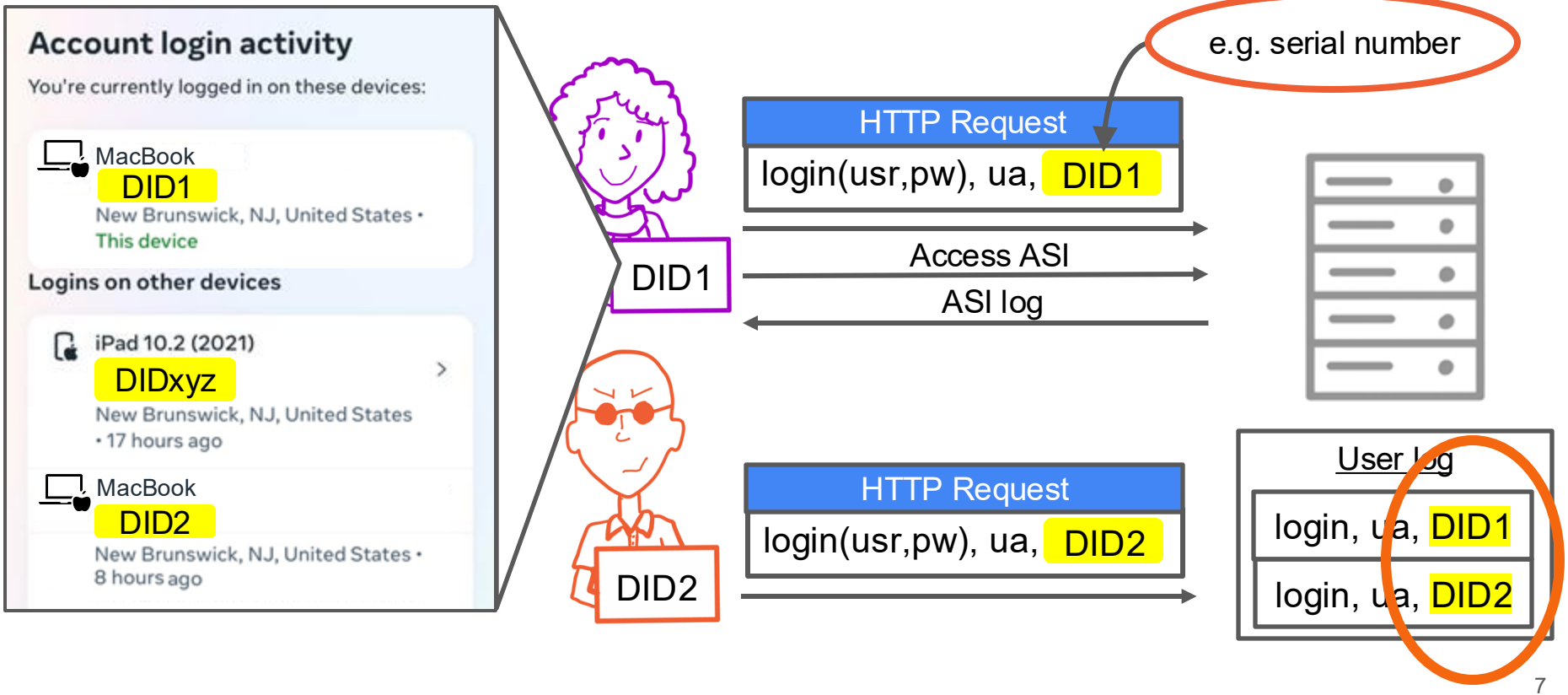
Straw approach: Adding reliable OS-provided information



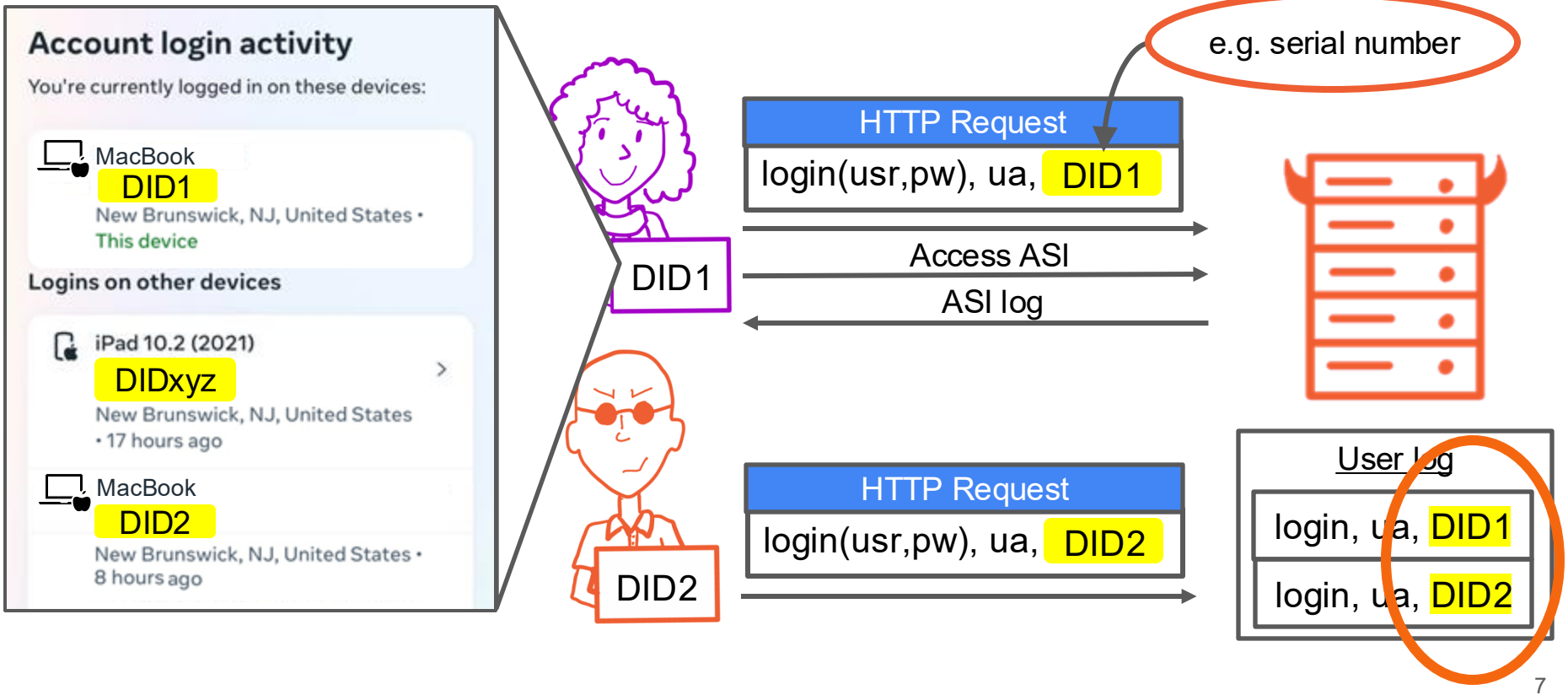
Straw approach: Adding reliable OS-provided information



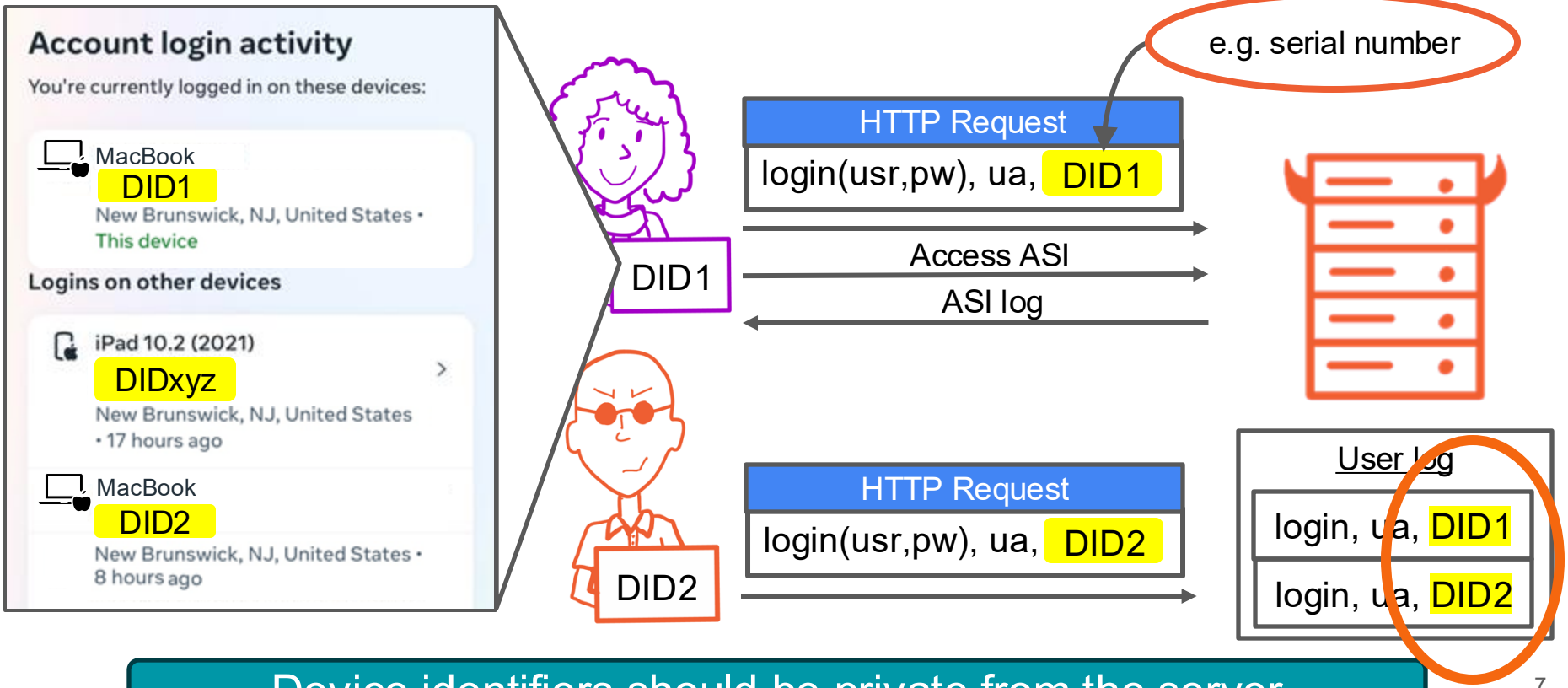
Straw approach: Adding reliable OS-provided information



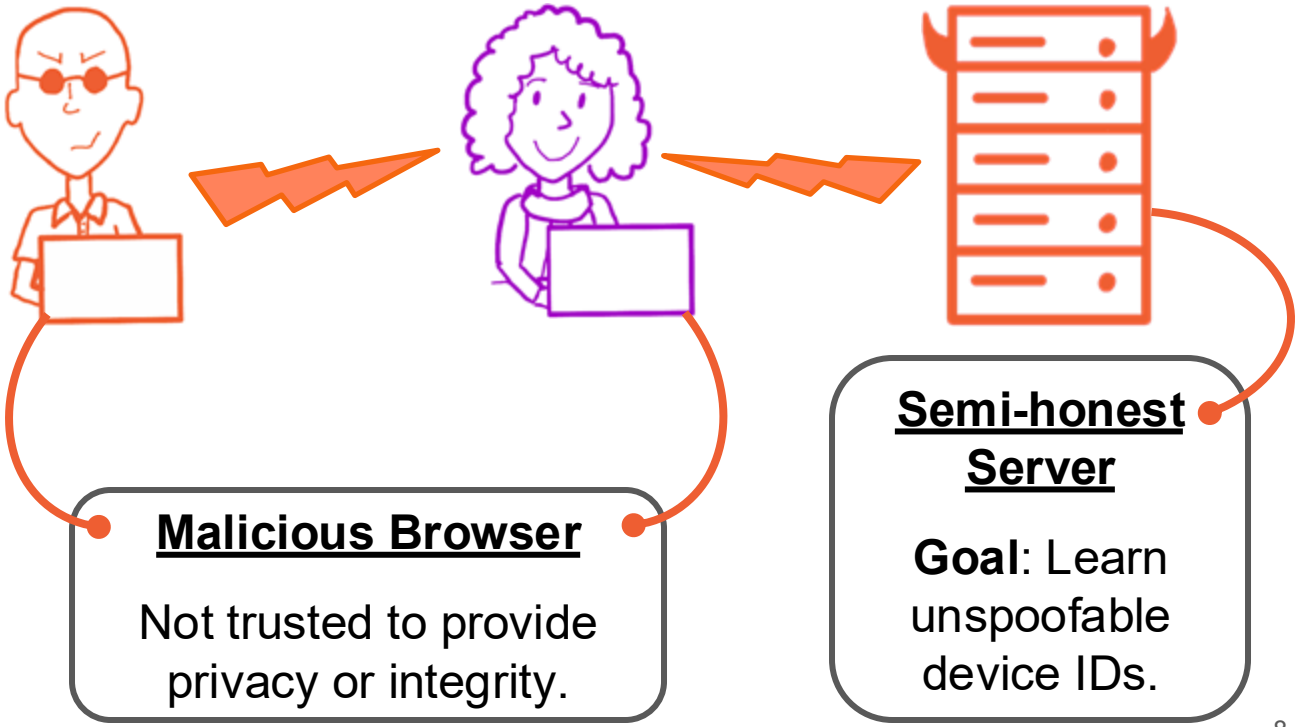
Straw approach: Adding reliable OS-provided information



Straw approach: Adding reliable OS-provided information



Providing attribution and privacy

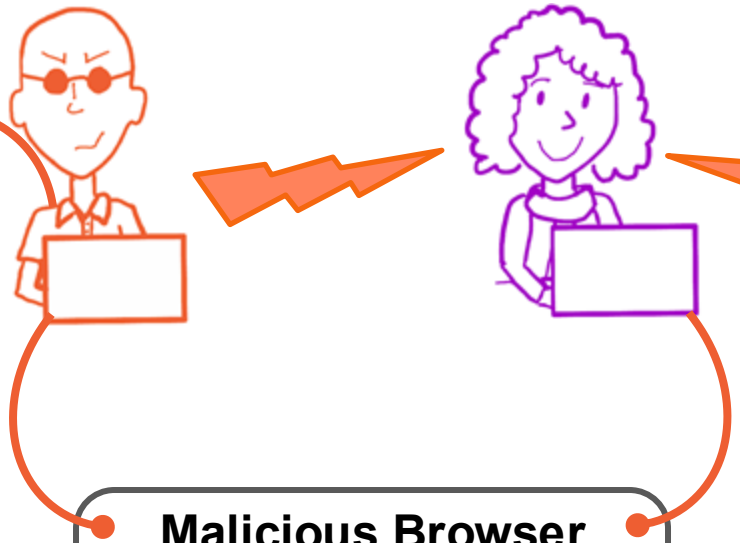


Providing attribution and privacy

Malicious User

Goal: Tamper with/circumvent log entries

*Can authenticate as the user.



Malicious Browser

Not trusted to provide privacy or integrity.

Semi-honest Server

Goal: Learn unspoofable device IDs.

Providing attribution and privacy

Malicious User

Goal: Tamper with/circumvent log entries

*Can authenticate as the user.

Encrypt & sign in the OS

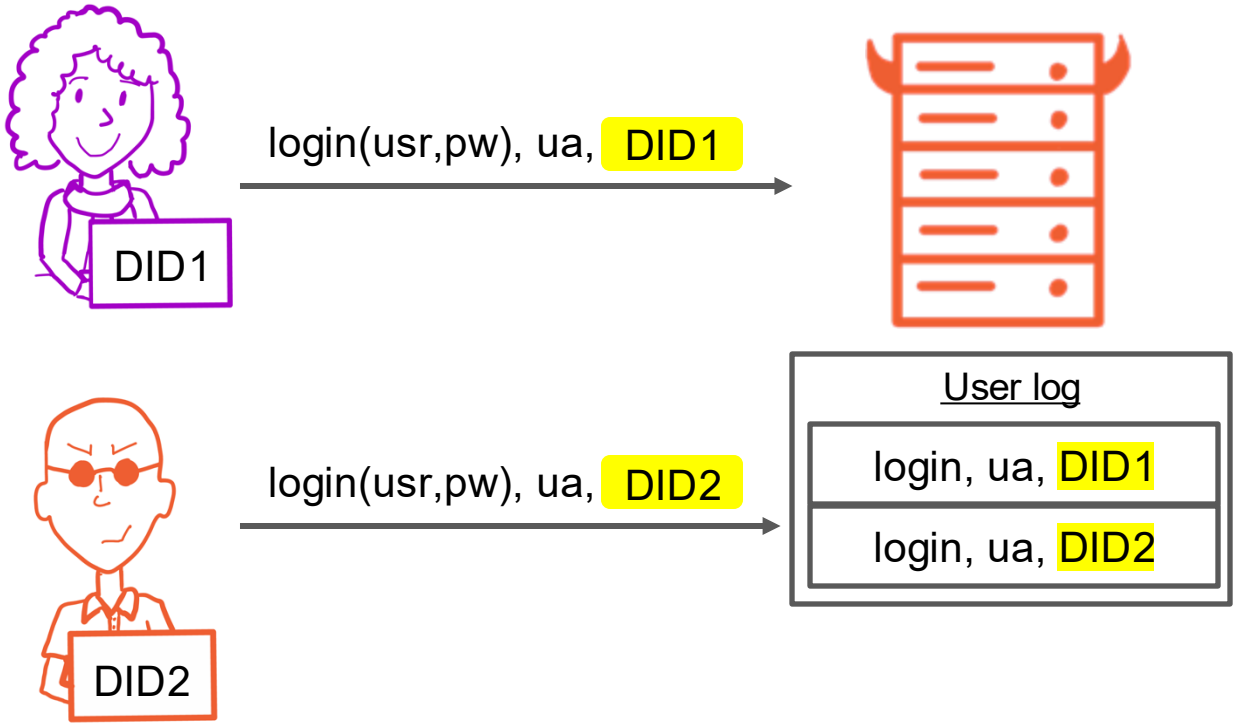
Malicious Browser

Not trusted to provide privacy or integrity.

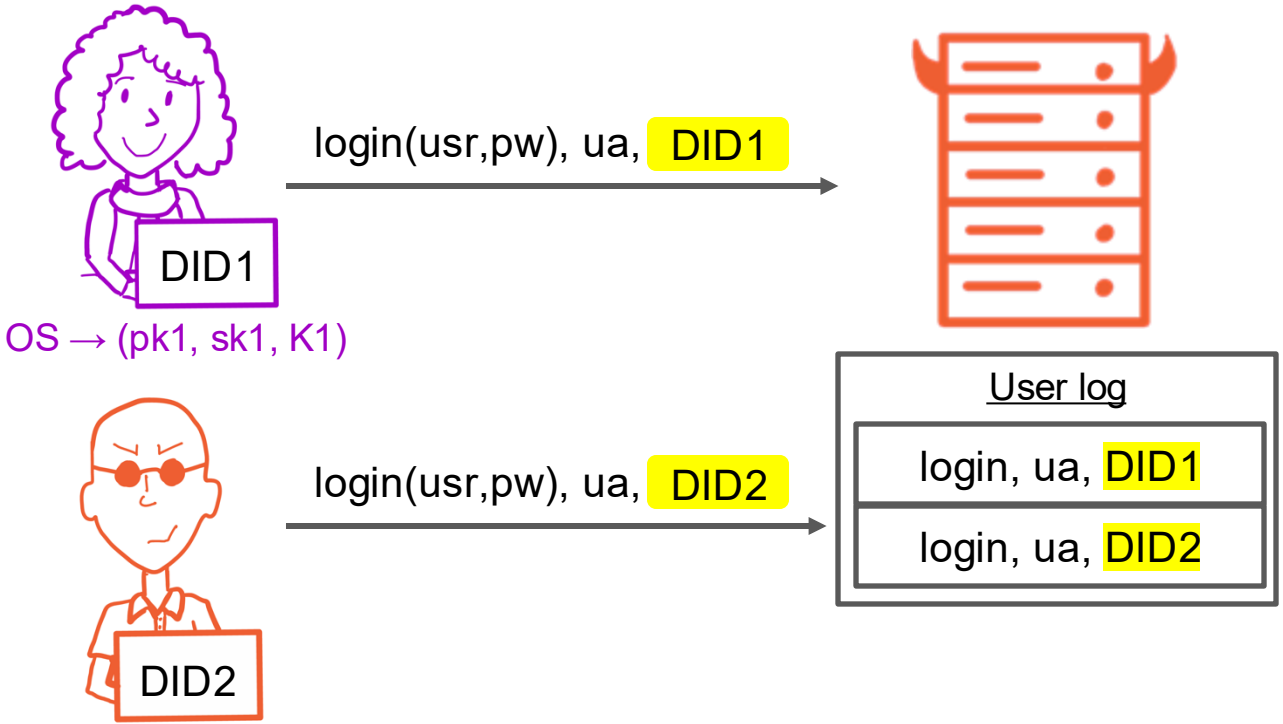
Semi-honest Server

Goal: Learn unspoofable device IDs.

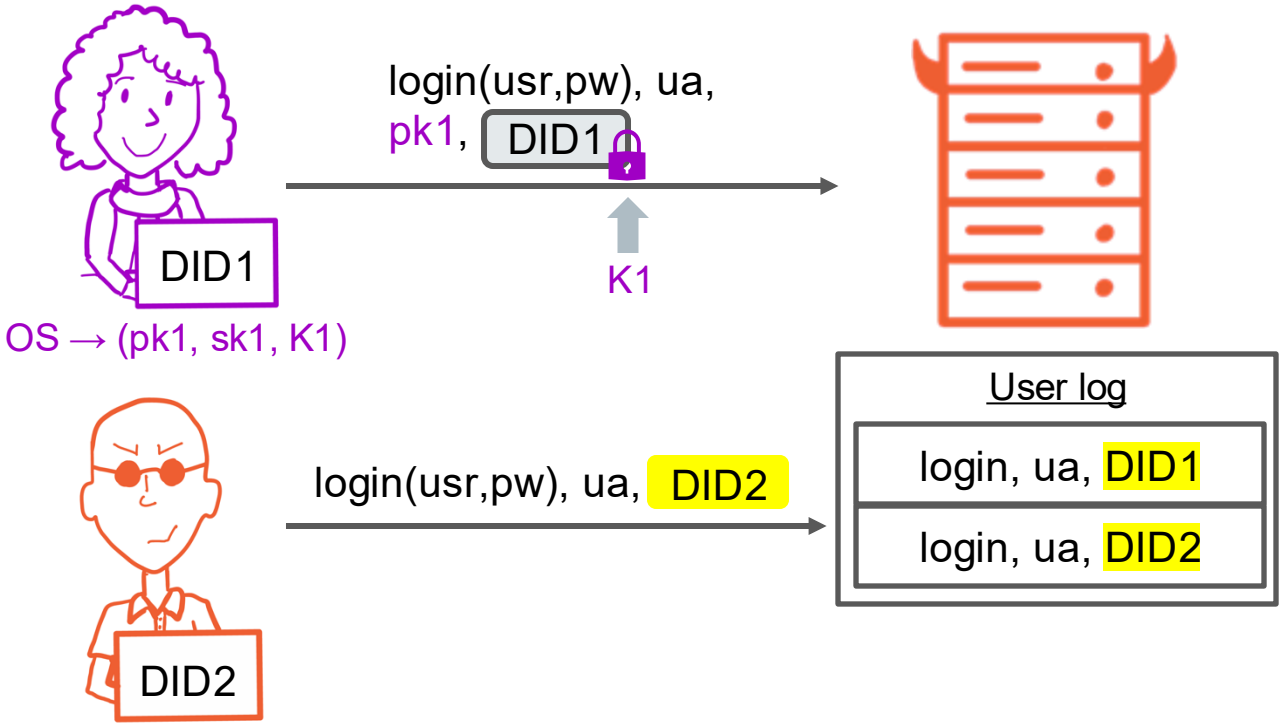
Client-side-encrypted access logging (CSAL)



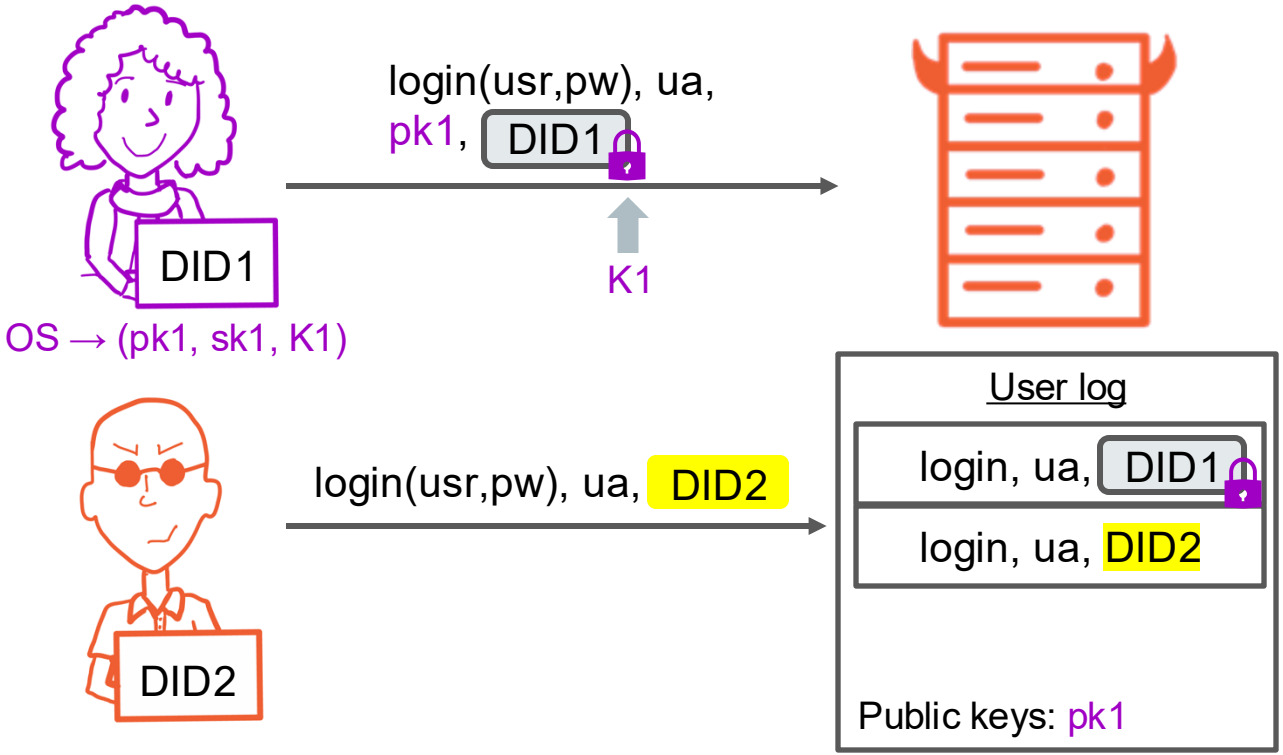
Client-side-encrypted access logging (CSAL)



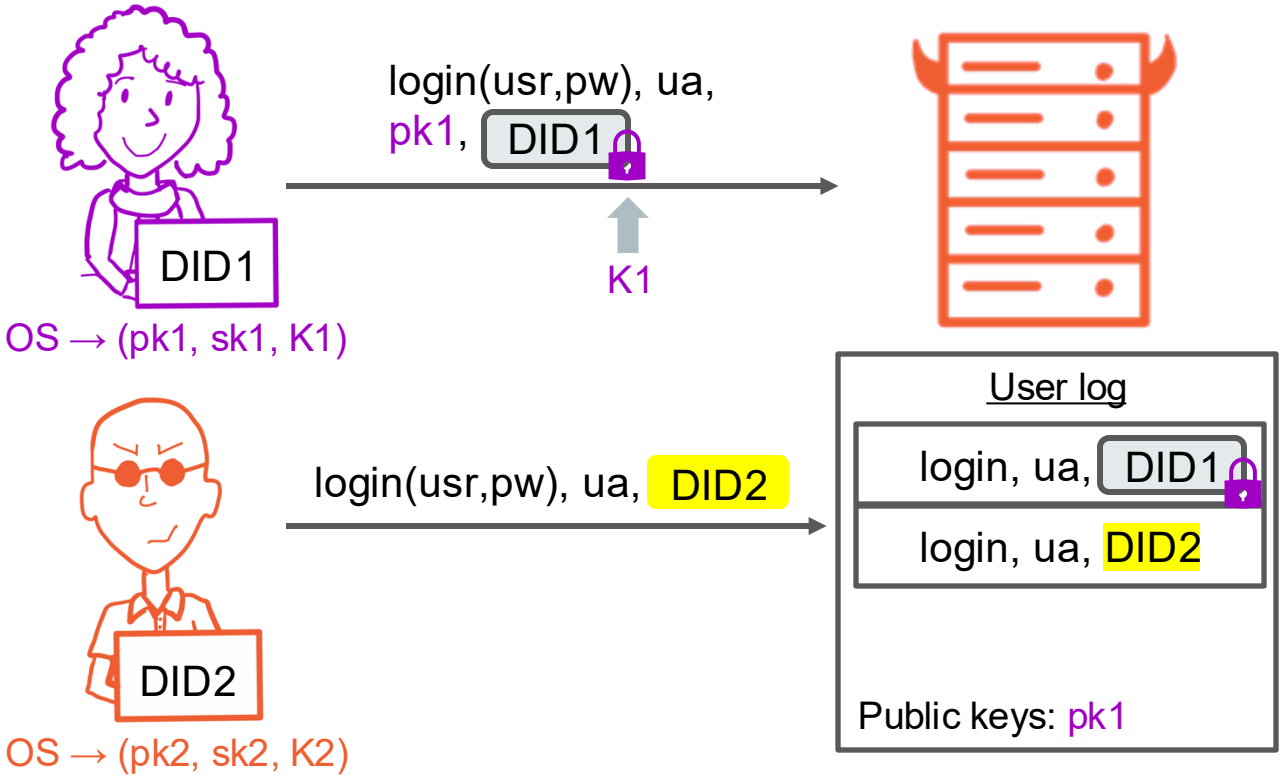
Client-side-encrypted access logging (CSAL)



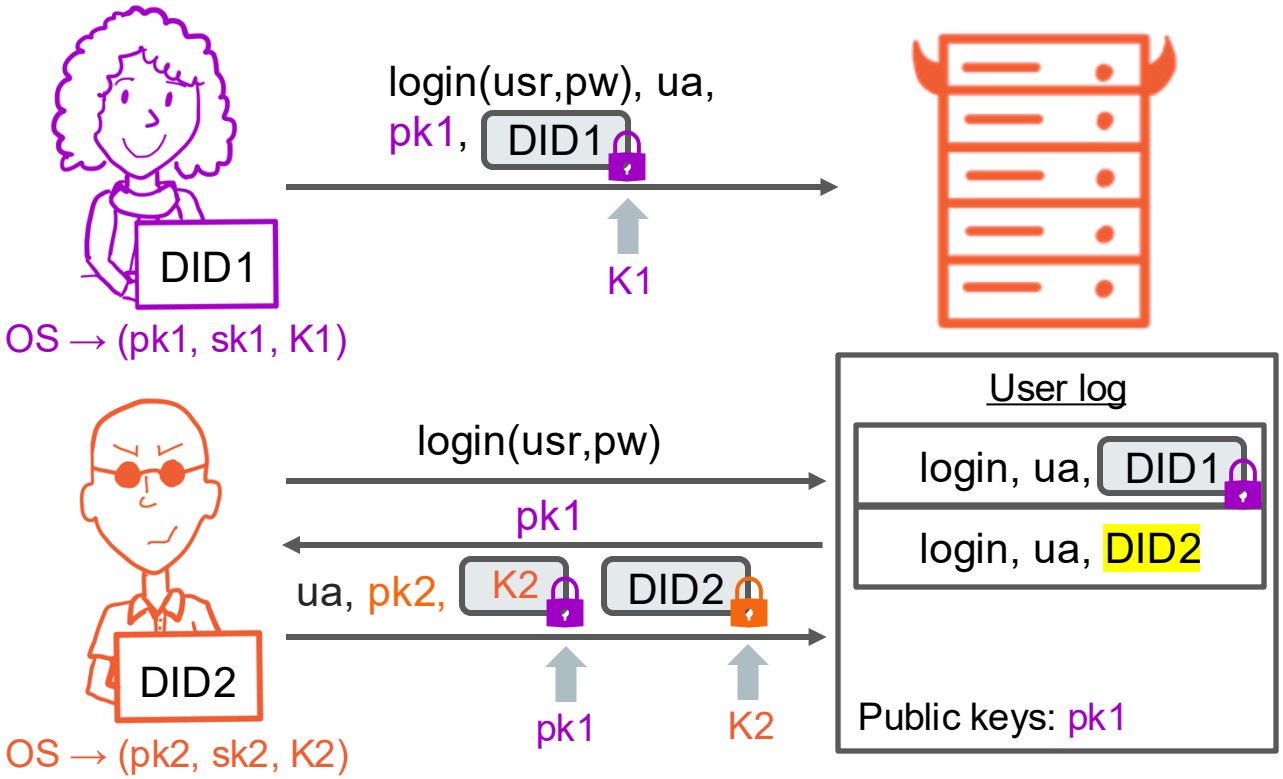
Client-side-encrypted access logging (CSAL)



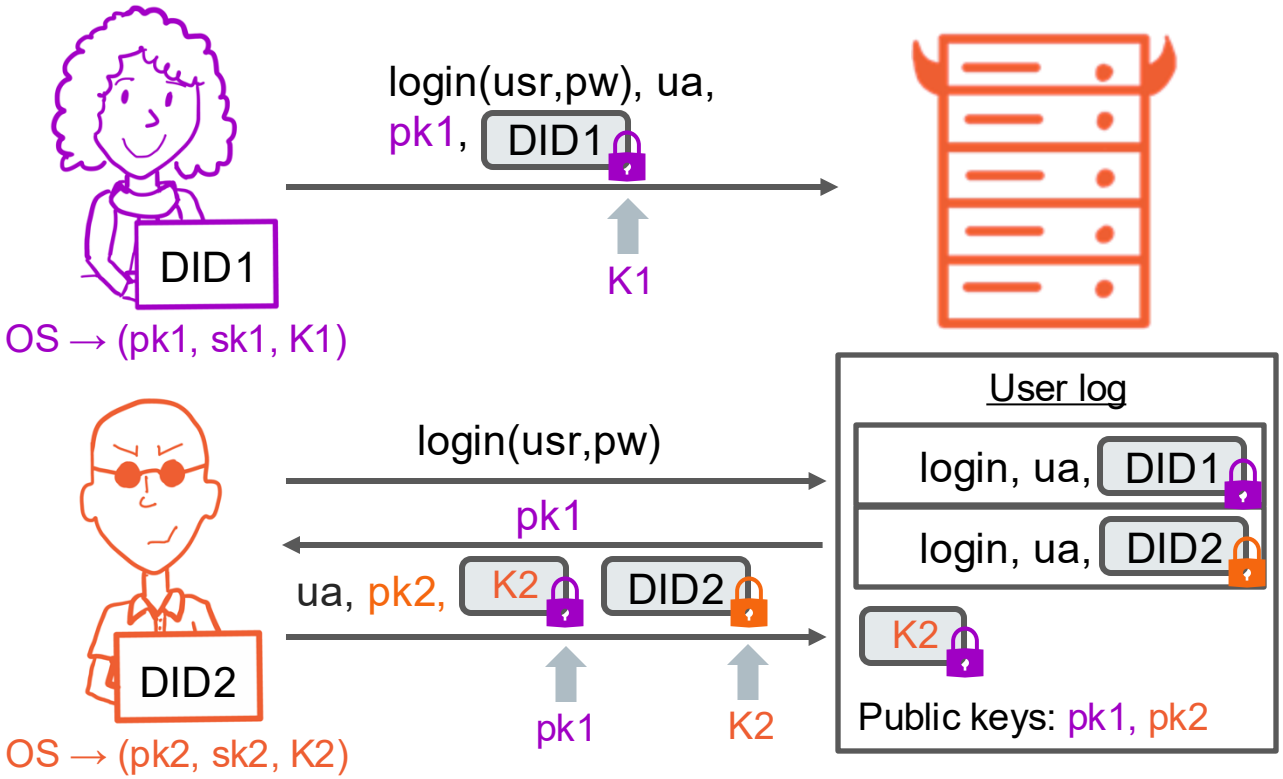
Client-side-encrypted access logging (CSAL)



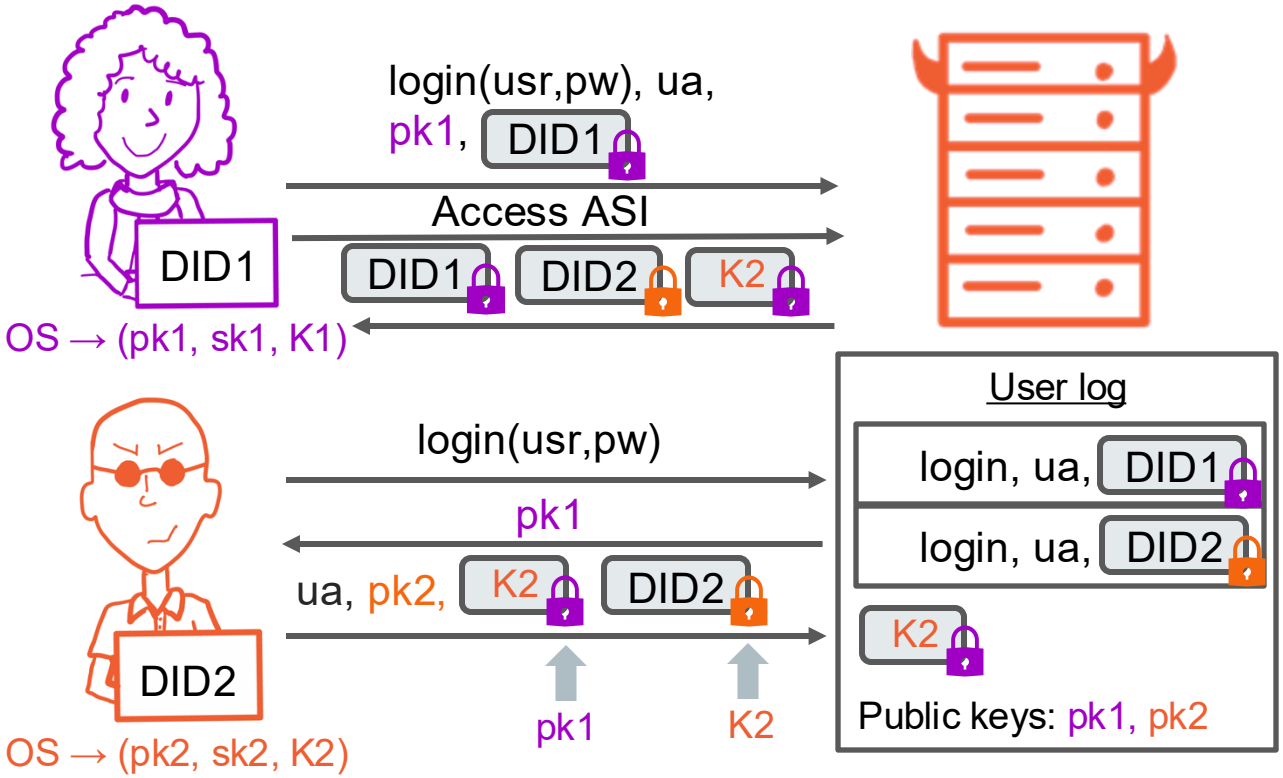
Client-side-encrypted access logging (CSAL)



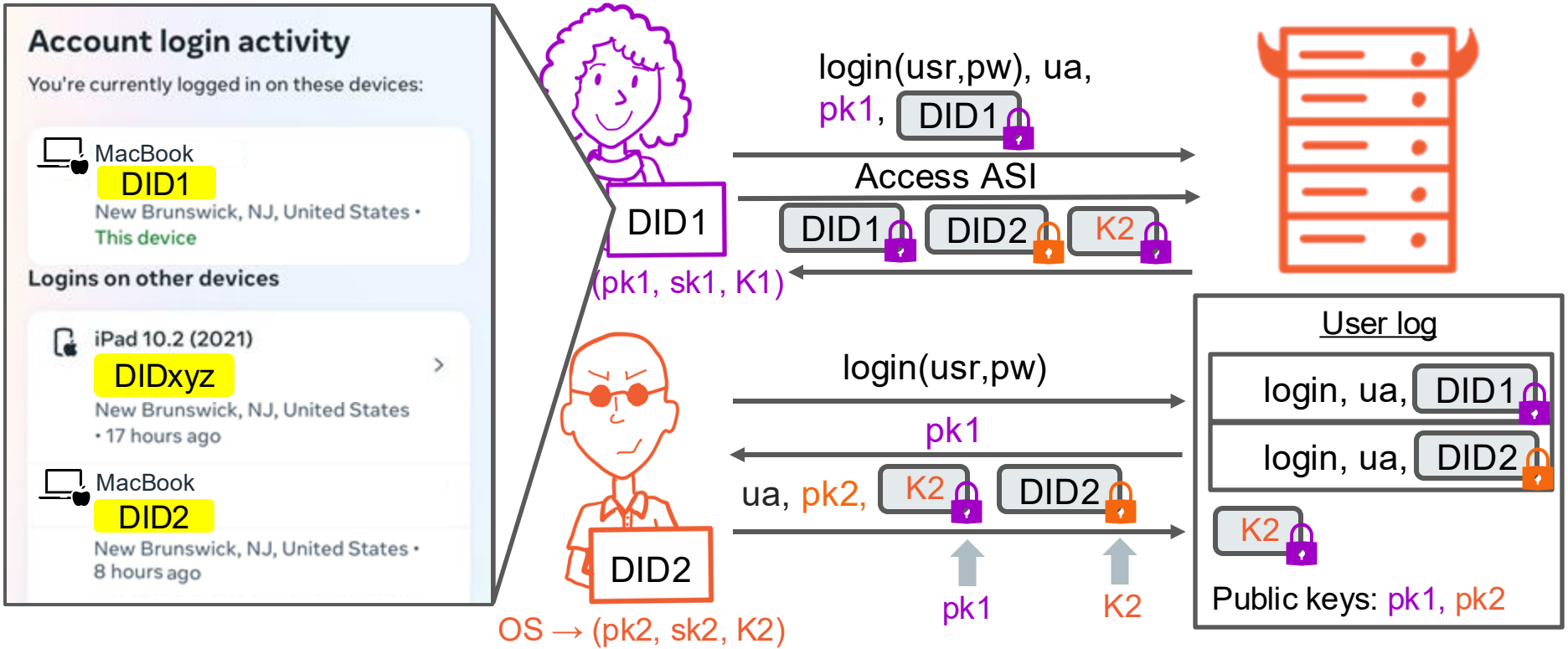
Client-side-encrypted access logging (CSAL)



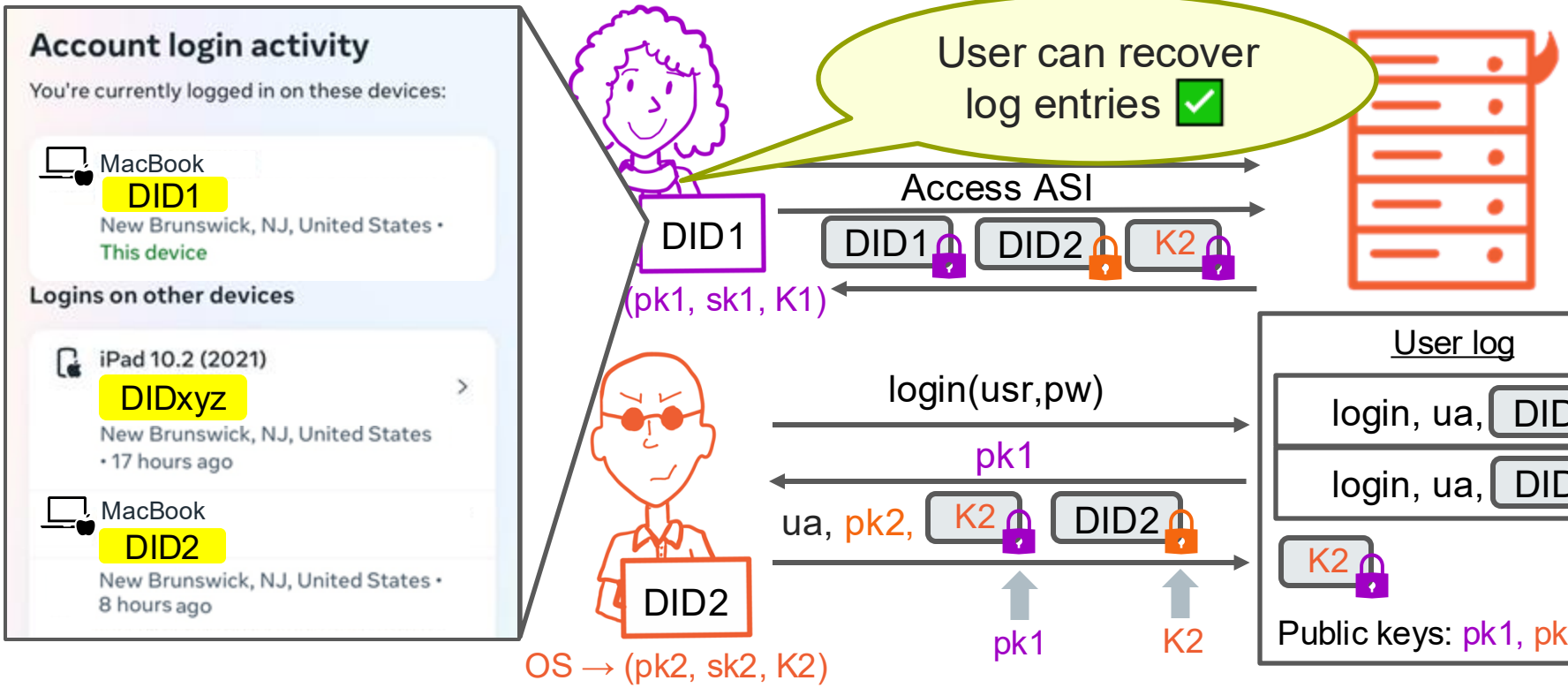
Client-side-encrypted access logging (CSAL)



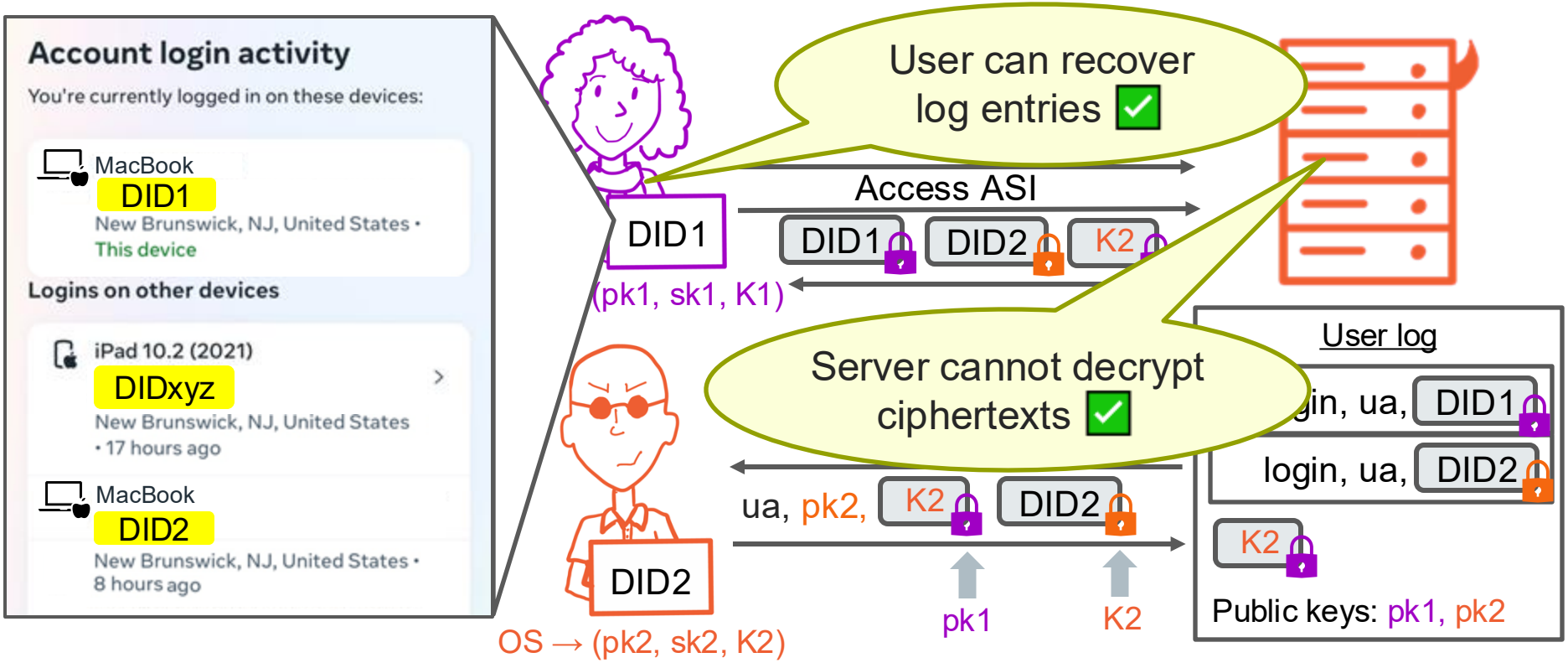
Client-side-encrypted access logging (CSAL)



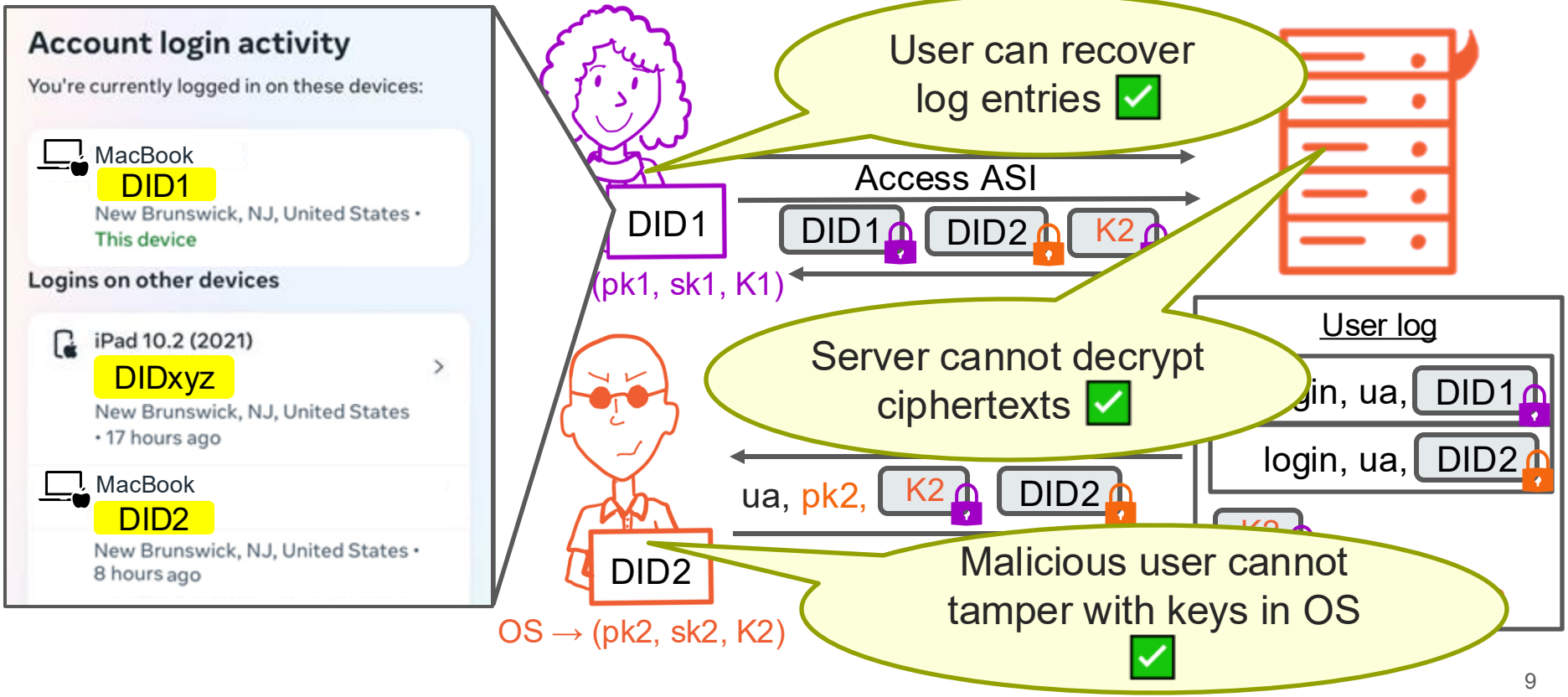
Client-side-encrypted access logging (CSAL)



Client-side-encrypted access logging (CSAL)

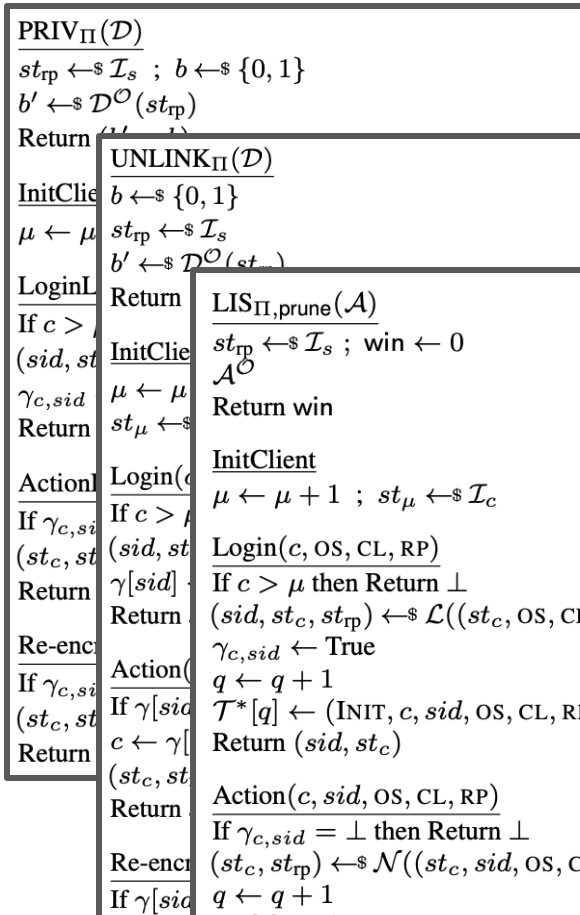


Client-side-encrypted access logging (CSAL)



Formal treatment of CSAL

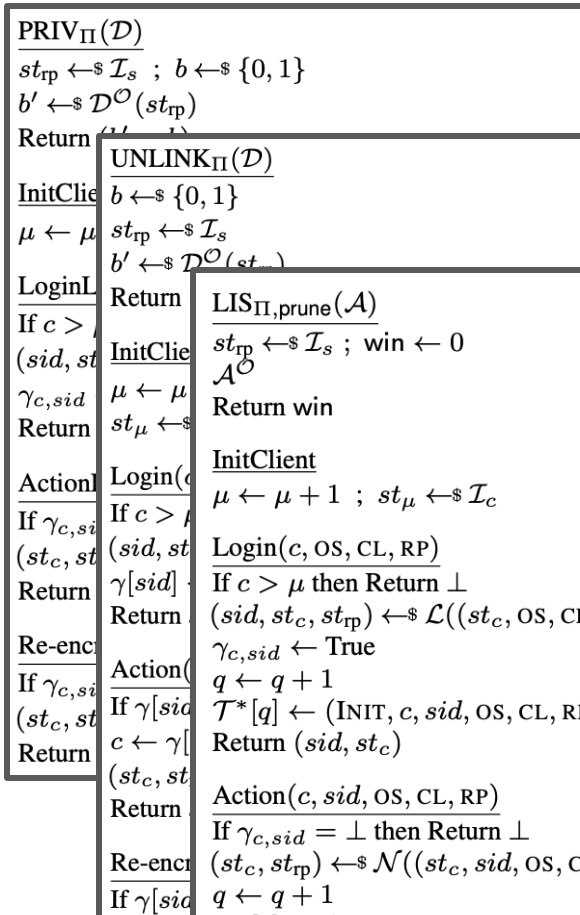
- **Log privacy** and **session unlinkability** against server
- **“Ability to access”** log entries



Formal treatment of CSAL

- **Log privacy** and **session unlinkability** against server
- **“Ability to access”** log entries

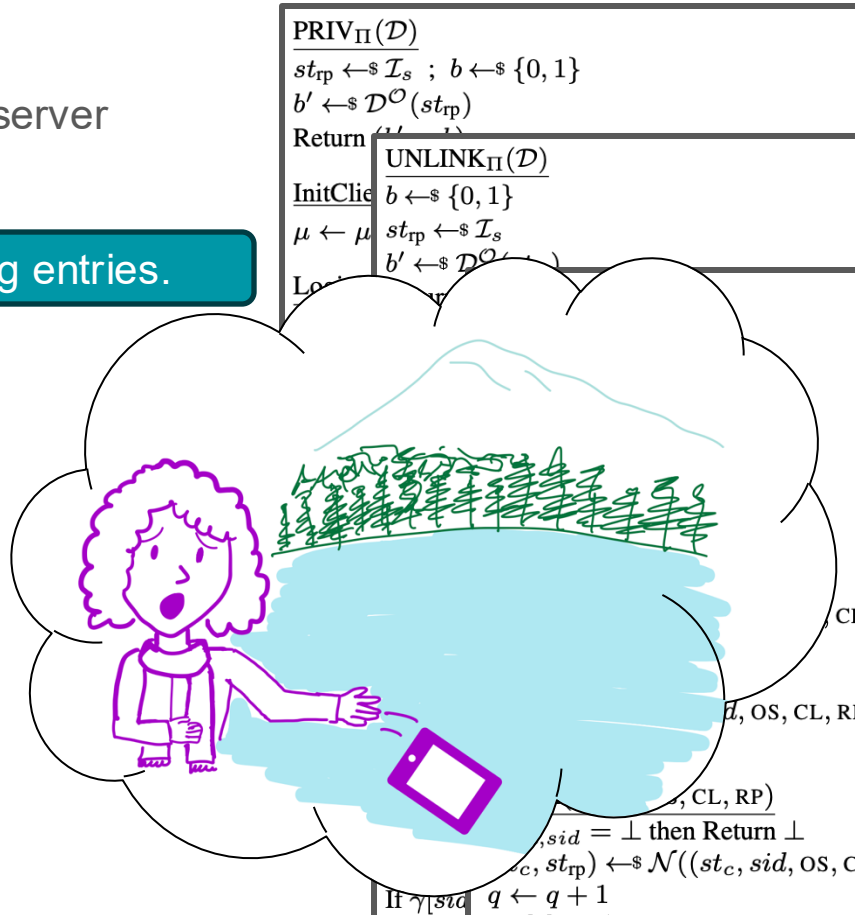
Theorem: Cannot have privacy and access all log entries.



Formal treatment of CSAL

- **Log privacy** and **session unlinkability** against server
- **“Ability to access”** log entries

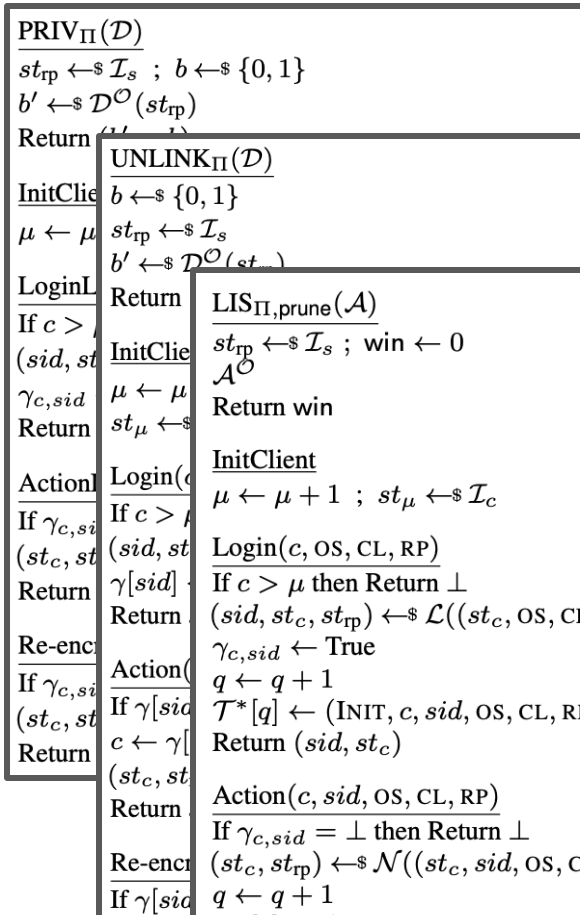
Theorem: Cannot have privacy and access all log entries.



Formal treatment of CSAL

- **Log privacy** and **session unlinkability** against server
- **“Ability to access”** log entries

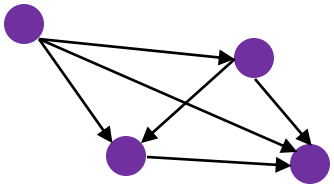
Theorem: Cannot have privacy and access all log entries.



Formal treatment of CSAL

- **Log privacy** and **session unlinkability** against server
- **“Ability to access”** log entries

Theorem: Cannot have privacy and access all log entries.



- Define reachability graph to characterize ciphertext access

$\underline{\text{PRIV}_{\Pi}(\mathcal{D})}$
$st_{\text{rp}} \leftarrow\!\$\ \mathcal{I}_s\ ;\ b \leftarrow\!\$\ \{0,1\}$
$b' \leftarrow\!\$\ \mathcal{D}^{\mathcal{O}}(st_{\text{rp}})$
Return

$\underline{\text{UNLINK}_{\Pi}(\mathcal{D})}$
$b \leftarrow\!\$\ \{0,1\}$
$st_{\text{rp}} \leftarrow\!\$\ \mathcal{I}_s$
Return

$\underline{\text{LIS}_{\Pi,\text{prune}}(\mathcal{A})}$
$st_{\text{rp}} \leftarrow\!\$\ \mathcal{I}_s\ ;\ \text{win} \leftarrow 0$
$\mathcal{A}^{\mathcal{O}}$
Return win

$\underline{\text{InitClient}}$
$\mu \leftarrow \mu + 1\ ;\ st_{\mu} \leftarrow\!\$\ \mathcal{I}_c$

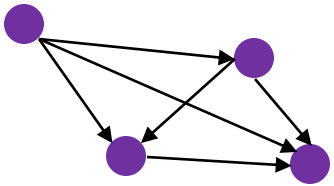
$\underline{\text{Login}(c, \text{OS}, \text{CL}, \text{RP})}$
If $c > \mu$ then Return $\perp$
$(sid, st_c, st_{\text{rp}}) \leftarrow\!\$\ \mathcal{L}((st_c, \text{OS}, \text{C}$
$\gamma_{c,sid} \leftarrow \text{True}$
$q \leftarrow q + 1$
$\mathcal{T}^*[q] \leftarrow (\text{INIT}, c, sid, \text{OS}, \text{CL}, \text{R}$
Return (sid, st_c)

$\underline{\text{Action}(c, sid, \text{OS}, \text{CL}, \text{RP})}$
If $\gamma_{c,sid} = \perp$ then Return $\perp$
$(st_c, st_{\text{rp}}) \leftarrow\!\$\ \mathcal{N}((st_c, sid, \text{OS}, \text{C}$
$q \leftarrow q + 1$

Formal treatment of CSAL

- **Log privacy** and **session unlinkability** against server
- **"Ability to access"** log entries

Theorem: Cannot have privacy and access all log entries.



- Define reachability graph to characterize ciphertext access

Theorem: Assuming secure encryption, our scheme achieves privacy, unlinkability, and log access determined by the reachability graph

$\underline{\text{PRIV}_{\Pi}(\mathcal{D})}$
$st_{\text{rp}} \leftarrow\$ \mathcal{I}_s$; $b \leftarrow\$ \{0,1\}$
$b' \leftarrow\$ \mathcal{D}^{\mathcal{O}}(st_{\text{rp}})$
Return

$\underline{\text{UNLINK}_{\Pi}(\mathcal{D})}$
$b \leftarrow\$ \{0,1\}$
$st_{\text{rp}} \leftarrow\$ \mathcal{I}_s$
Return

$\underline{\text{LIS}_{\Pi,\text{prune}}(\mathcal{A})}$
$st_{\text{rp}} \leftarrow\$ \mathcal{I}_s$; win $\leftarrow 0$
$\mathcal{A}^{\mathcal{O}}$
Return win

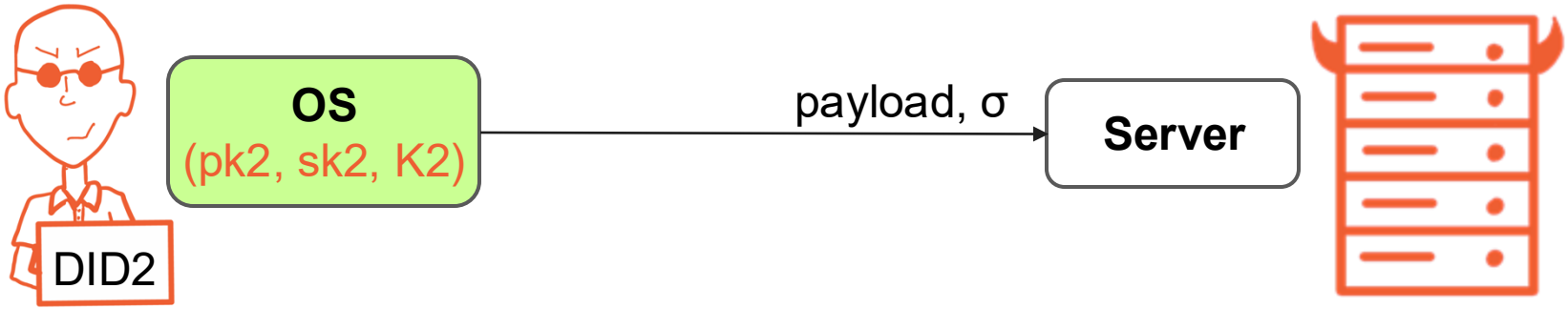
$\underline{\text{InitClient}}$
$\mu \leftarrow \mu + 1$; $st_{\mu} \leftarrow\$ \mathcal{I}_c$

$\underline{\text{Login}(c, \text{OS}, \text{CL}, \text{RP})}$
If $c > \mu$ then Return $\perp$
$(sid, st_c, st_{\text{rp}}) \leftarrow\$ \mathcal{L}((st_c, \text{OS}, \text{C}$
$\gamma_{c,sid} \leftarrow$ True
$q \leftarrow q + 1$
$\mathcal{T}^*[q] \leftarrow (\text{INIT}, c, sid, \text{OS}, \text{CL}, \text{R}$
Return (sid, st_c)

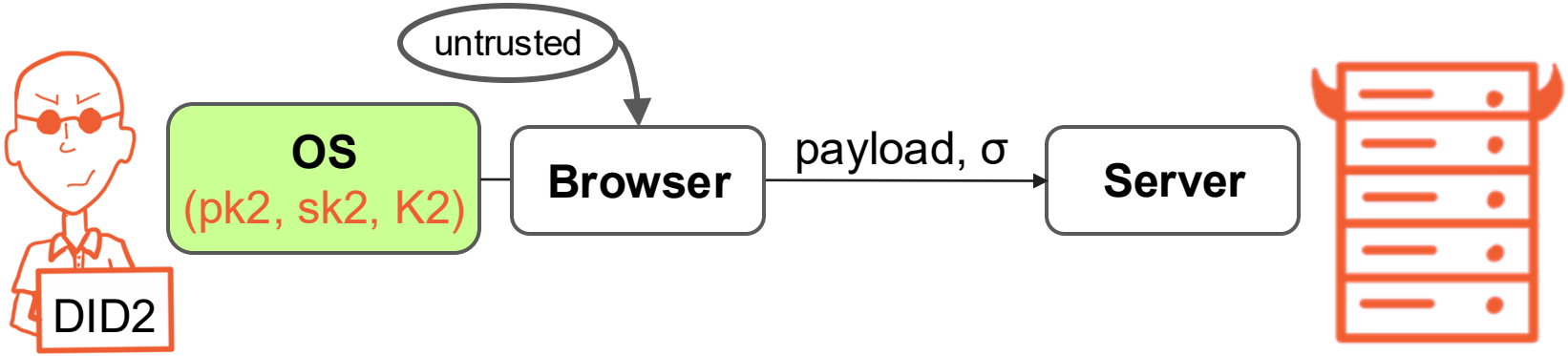
$\underline{\text{Action}(c, sid, \text{OS}, \text{CL}, \text{RP})}$
If $\gamma_{c,sid} = \perp$ then Return $\perp$
$(st_c, st_{\text{rp}}) \leftarrow\$ \mathcal{N}((st_c, sid, \text{OS}, \text{C}$
$q \leftarrow q + 1$

CSAL deployment architecture challenges

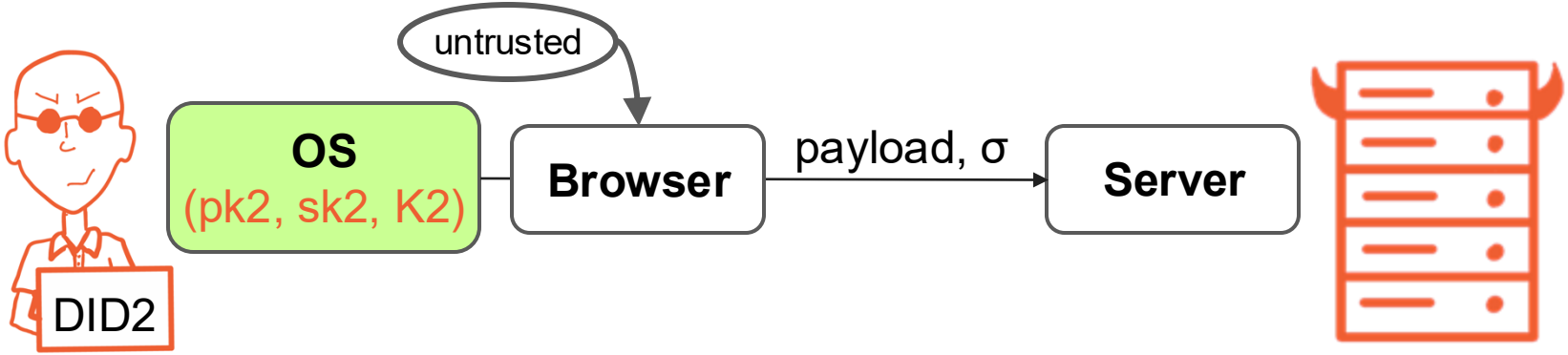
CSAL deployment architecture challenges



CSAL deployment architecture challenges

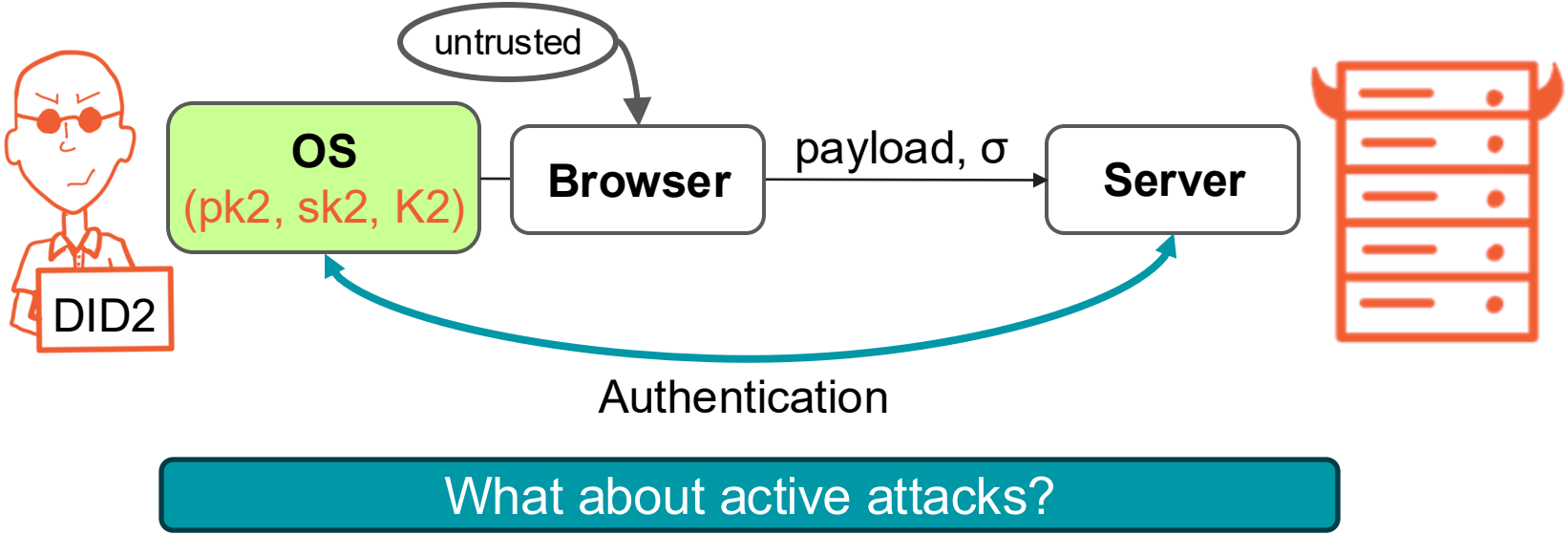


CSAL deployment architecture challenges



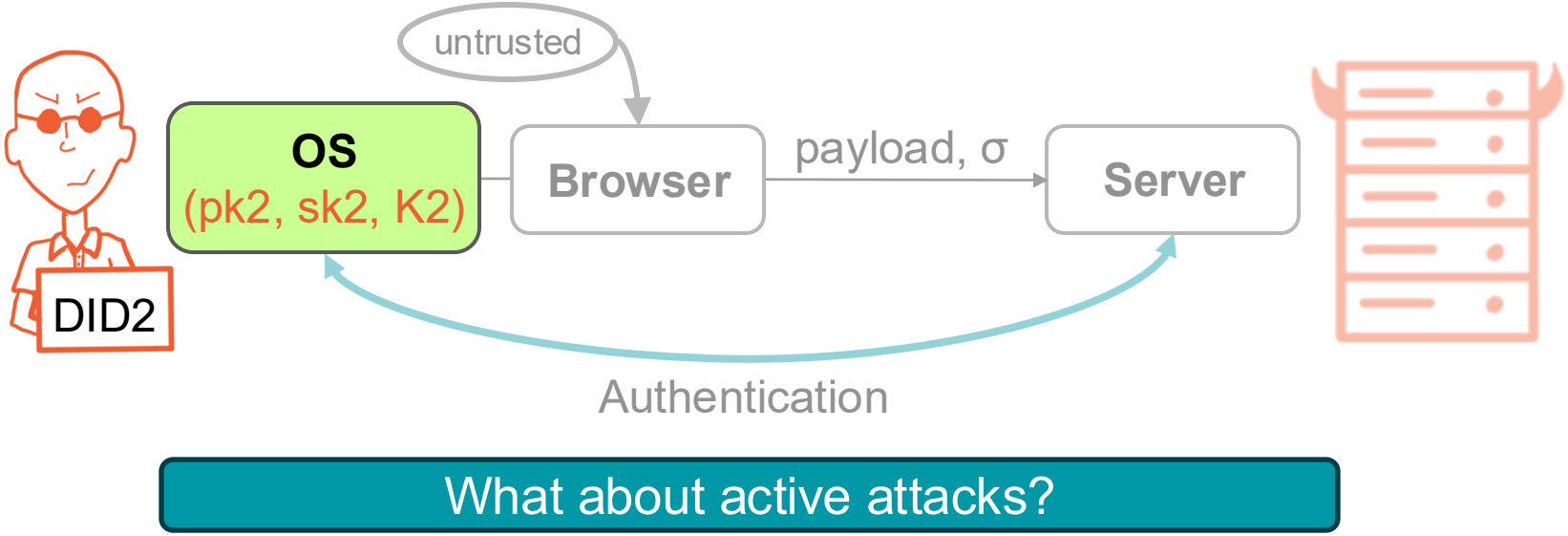
What about active attacks?

CSAL deployment architecture challenges



What about active attacks?

CSAL deployment architecture challenges



What about active attacks?

CSAL deployment architecture: OS authentication

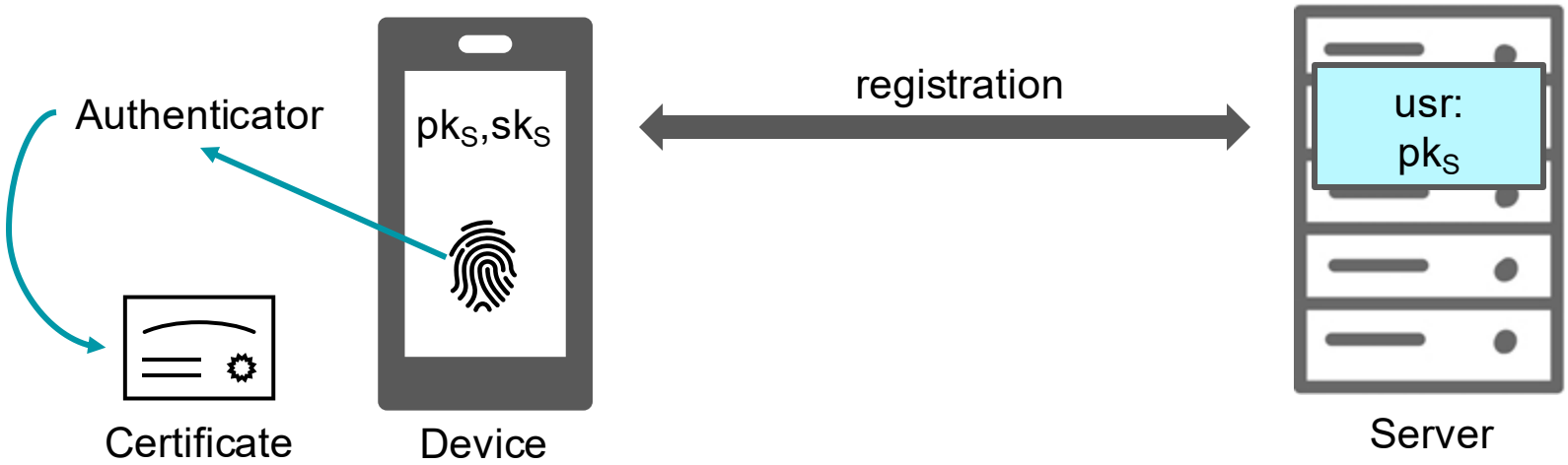
- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

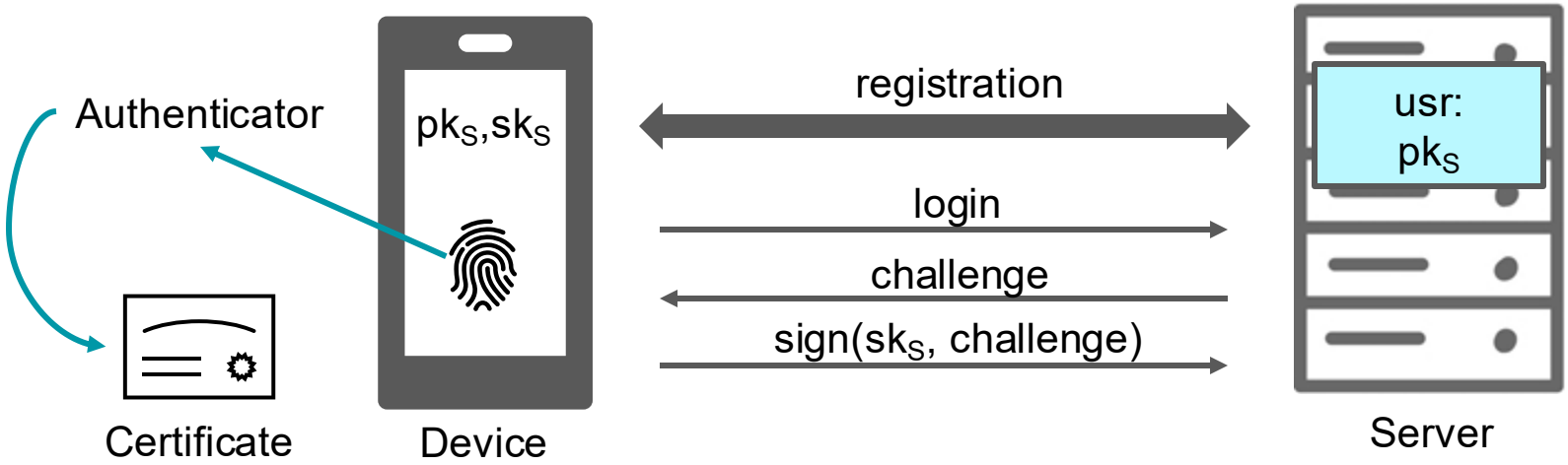
CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator



CSAL deployment architecture: OS authentication

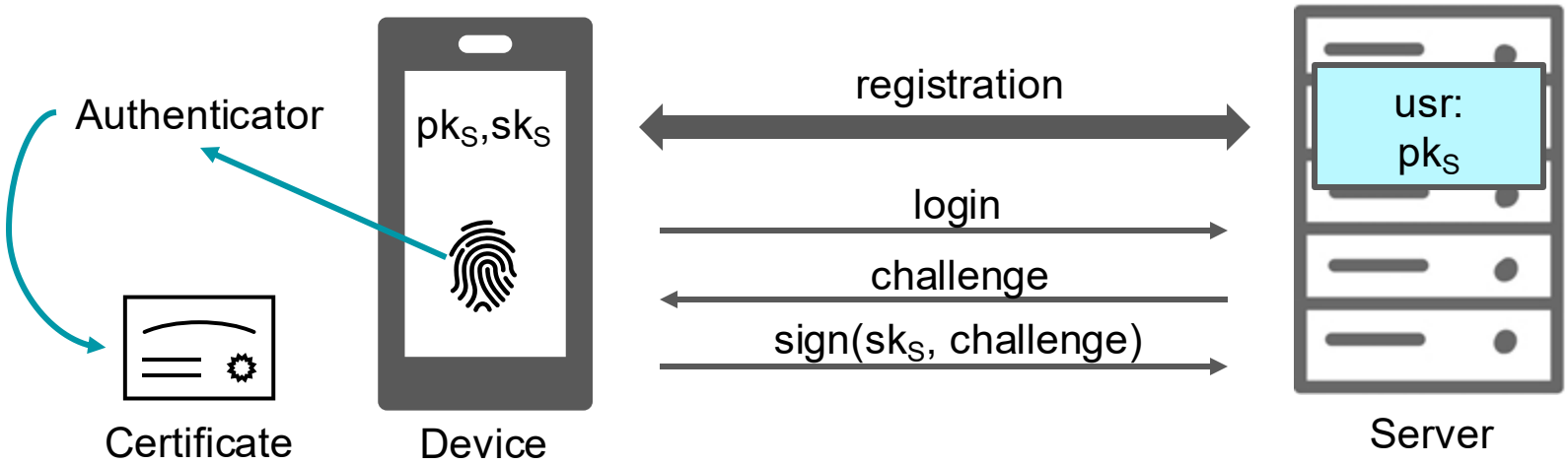
- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator



CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

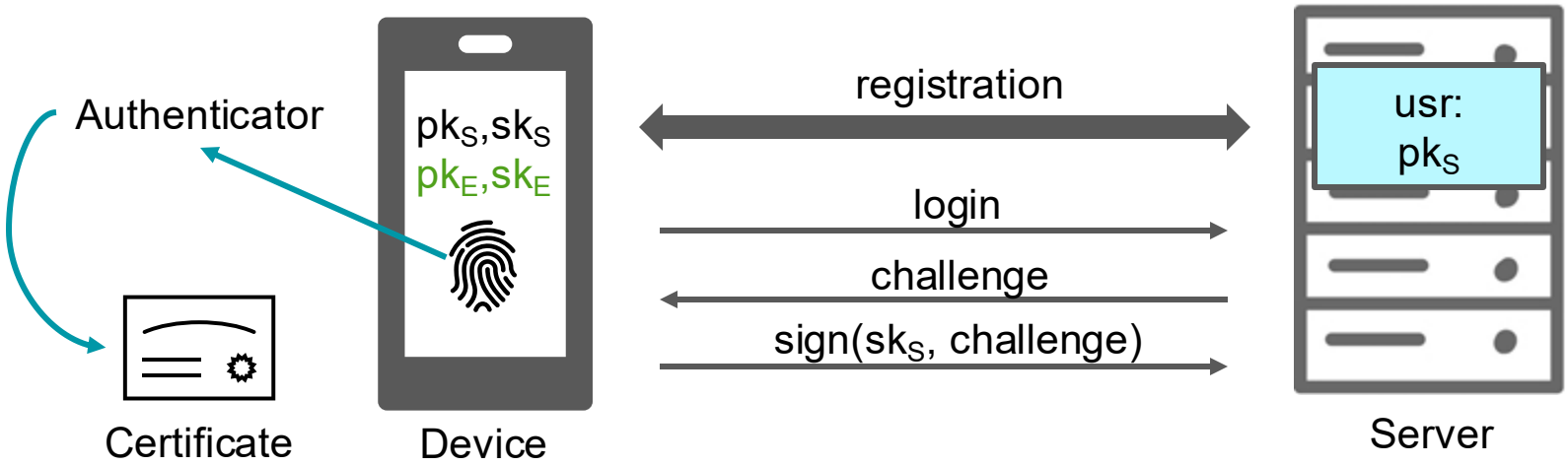
Why FIDO2?
- Collaboration with OS vendors
- Handles cryptographic keys
- Widely deployed



CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

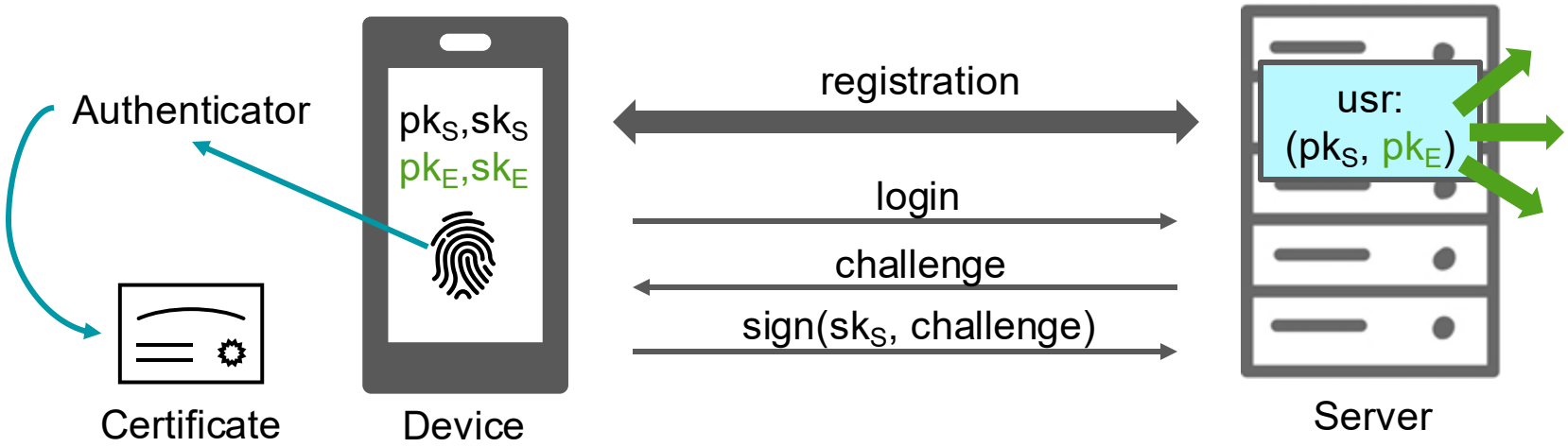
Why FIDO2?
- Collaboration with OS vendors
- Handles cryptographic keys
- Widely deployed



CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

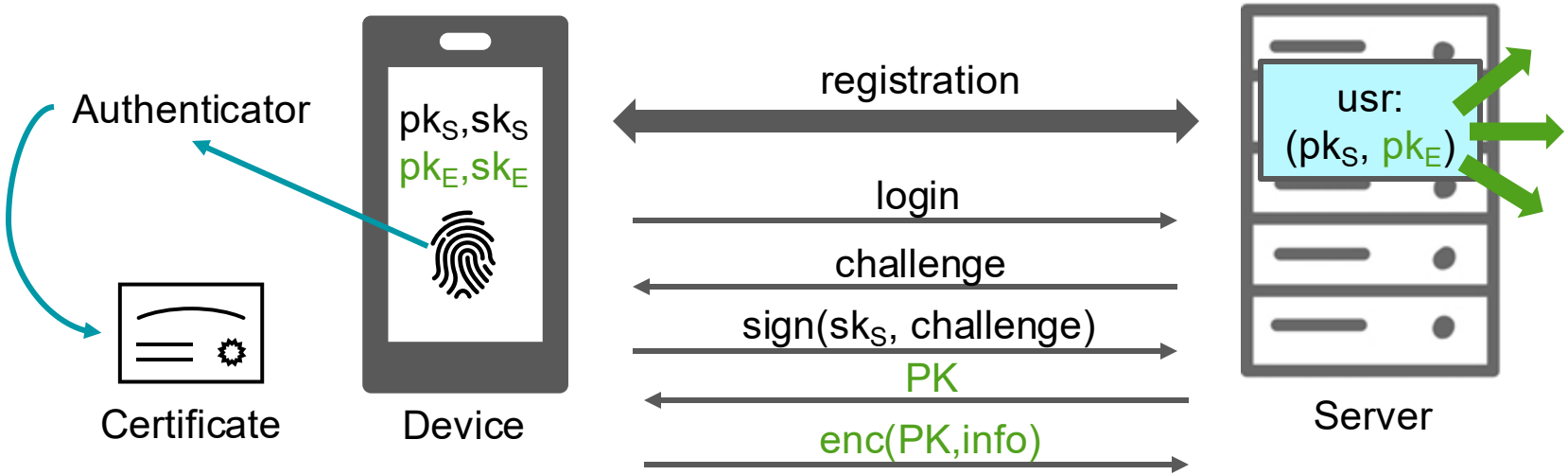
Why FIDO2?
- Collaboration with OS vendors
- Handles cryptographic keys
- Widely deployed



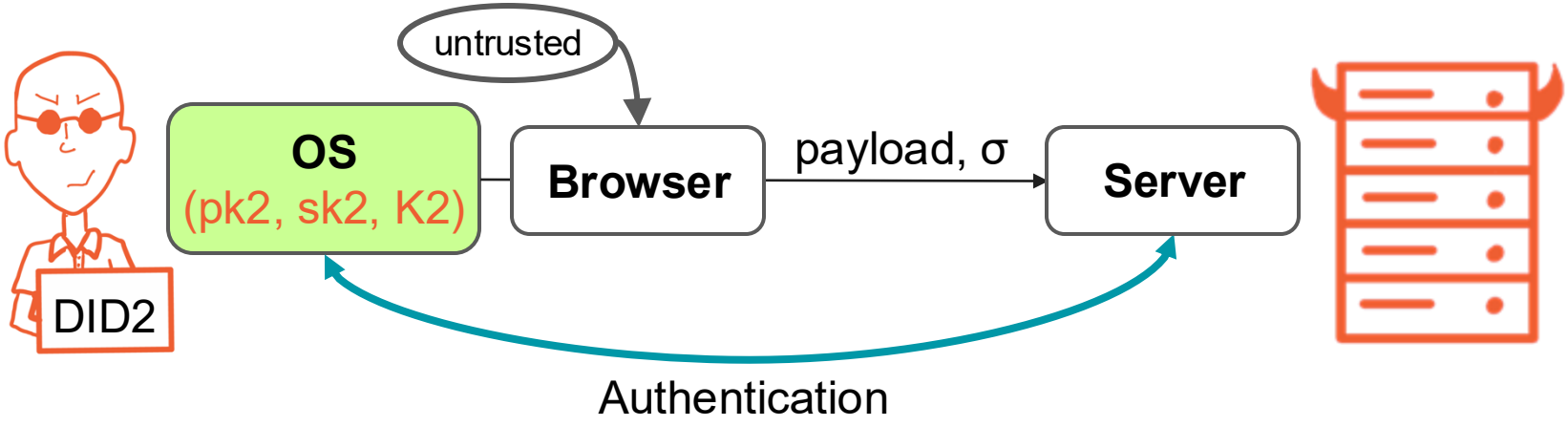
CSAL deployment architecture: OS authentication

- Based on the **FIDO2 framework**
  - Specifications for device authentication
  - WebAuthn → Passkeys
    - API to register and authenticate users using an authenticator

Why FIDO2?
- Collaboration with OS vendors
- Handles cryptographic keys
- Widely deployed



CSAL deployment architecture challenges

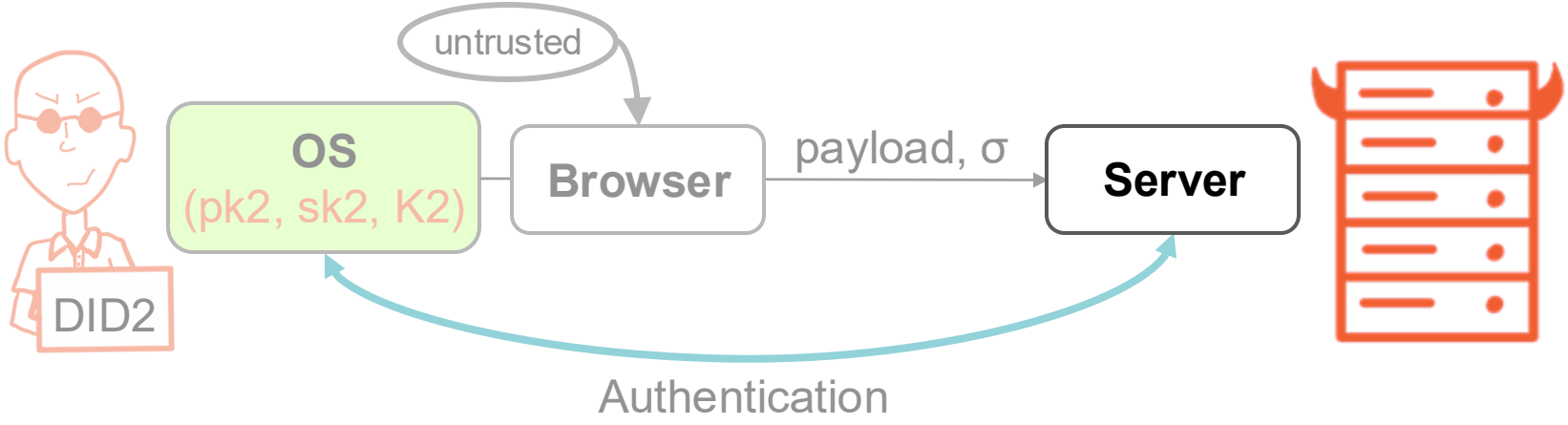


What about active attacks?

OS Authentication

FIDO2-style certificates

CSAL deployment architecture challenges

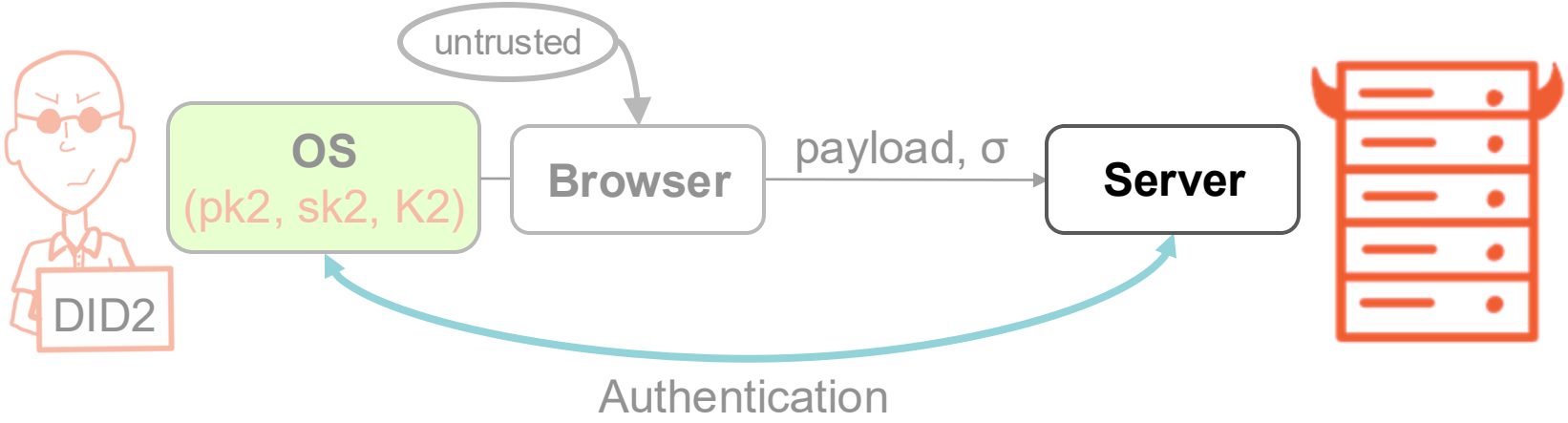


What about active attacks?

OS Authentication

FIDO2-style certificates

CSAL deployment architecture challenges



What about active attacks?

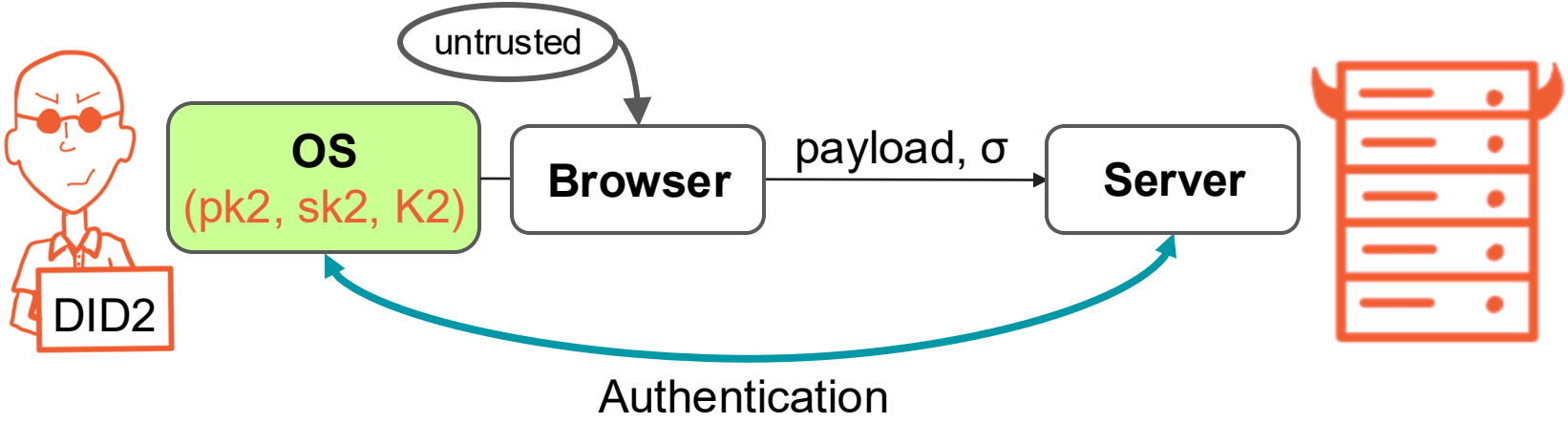
OS Authentication

FIDO2-style certificates

Server PKI

Infrastructure similar to TLS

CSAL deployment architecture challenges



What about active attacks?

OS Authentication

FIDO2-style certificates

Server PKI

Infrastructure similar to TLS

✅ Browser MiTM

❌ Server MiTM

Proof of concept

- Prototype Python implementation of a server, client, and encryptor
  - Login:
    - Bandwidth: ~5KB per single session login
    - Latency: 146ms for single session login
  - ASI log retrieval:
    - Bandwidth: ~2.6KB for one session
    - Latency: 0.63ms for single entry

Proof of concept

- Prototype Python implementation of a server, client, and encryptor
  - Login:
    - Bandwidth: ~5KB per single session login
    - Latency: 146ms for single session login
  - ASI log retrieval:
    - Bandwidth: ~2.6KB for one session
    - Latency: 0.63ms for single entry

Performance-wise, the system is practical

Future work

- Deployment
  - Collaboration between OS vendors and browsers and other clients
  - Similar to FIDO2
- Formalization against malicious server
- ASIs potential risks
  - Risks of user monitoring by abuser
- ASIs UX
  - What type of information should we include in ASIs?

Encrypted Access Logging for Online Accounts: Device Attributions without Device Tracking

Takeaways:

- ASIs are important
- Trade-offs between integrity, privacy, and full access to logs
- OS signed identifiers prevent other users from tampering with logs
- Encryption ensures privacy from server
- Open challenges surrounding architecture that requires collaboration between OS vendors and platforms

Carolina Ortega Pérez
carolina@cs.cornell.edu

Alaa Daffalla
alaadaffalla@cs.cornell.edu

Thomas Ristenpart
ristenpart@cornell.edu

Paper:
https://bit.ly/452AiTa