Securing the Foundations of Verifiable Credential Ecosystems

Daniel Fett, TDI 2024

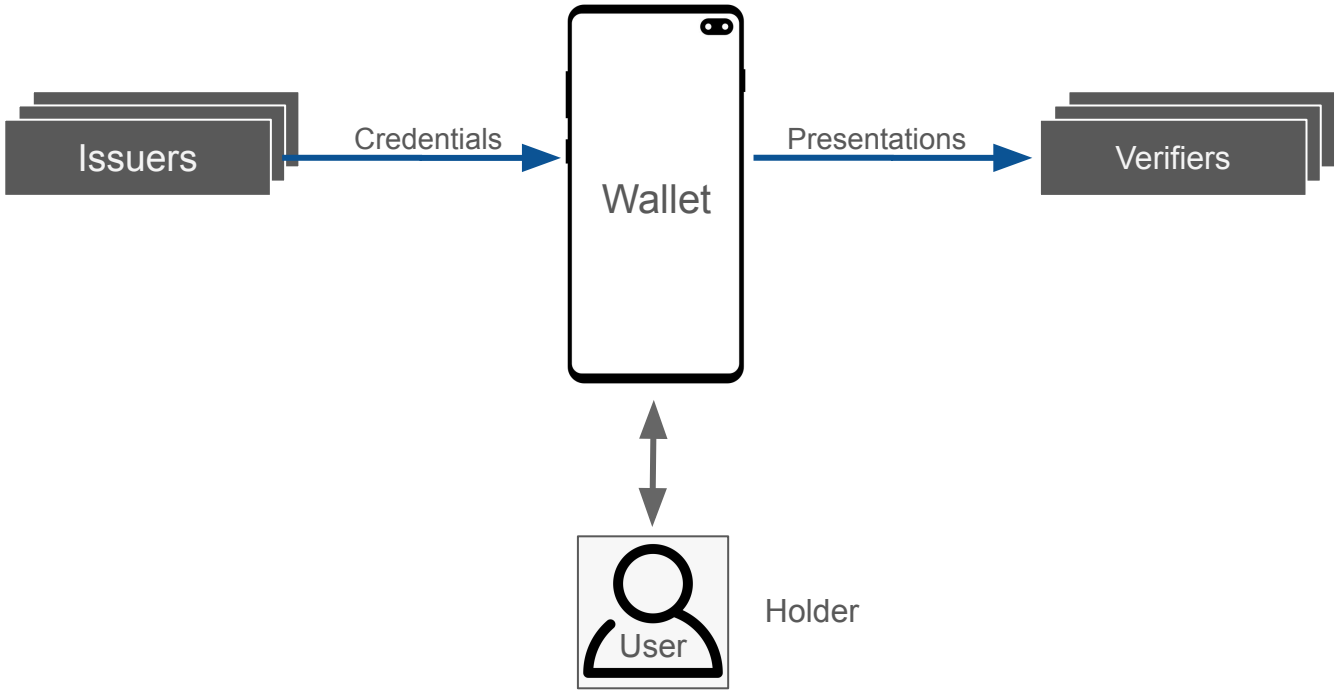
Daniel Fett

Security and Standardization Expert

Authlete

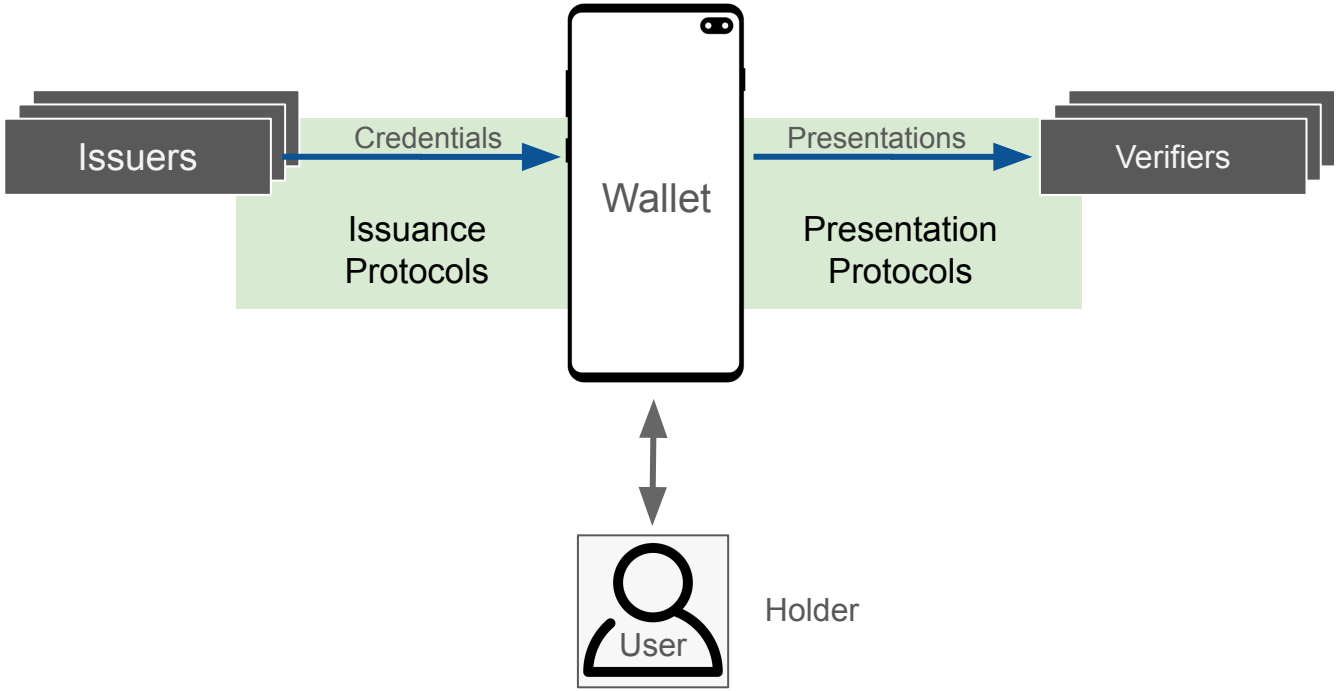
- PhD on formal security analysis of web protocols
- Experience in large-scale open banking ecosystems (yes.com)
- Contributing to Standards in the IETF and OpenID Foundation for Authlete
- Also happens to run the OAuth Security Workshop

Verifiable Credential Ecosystems



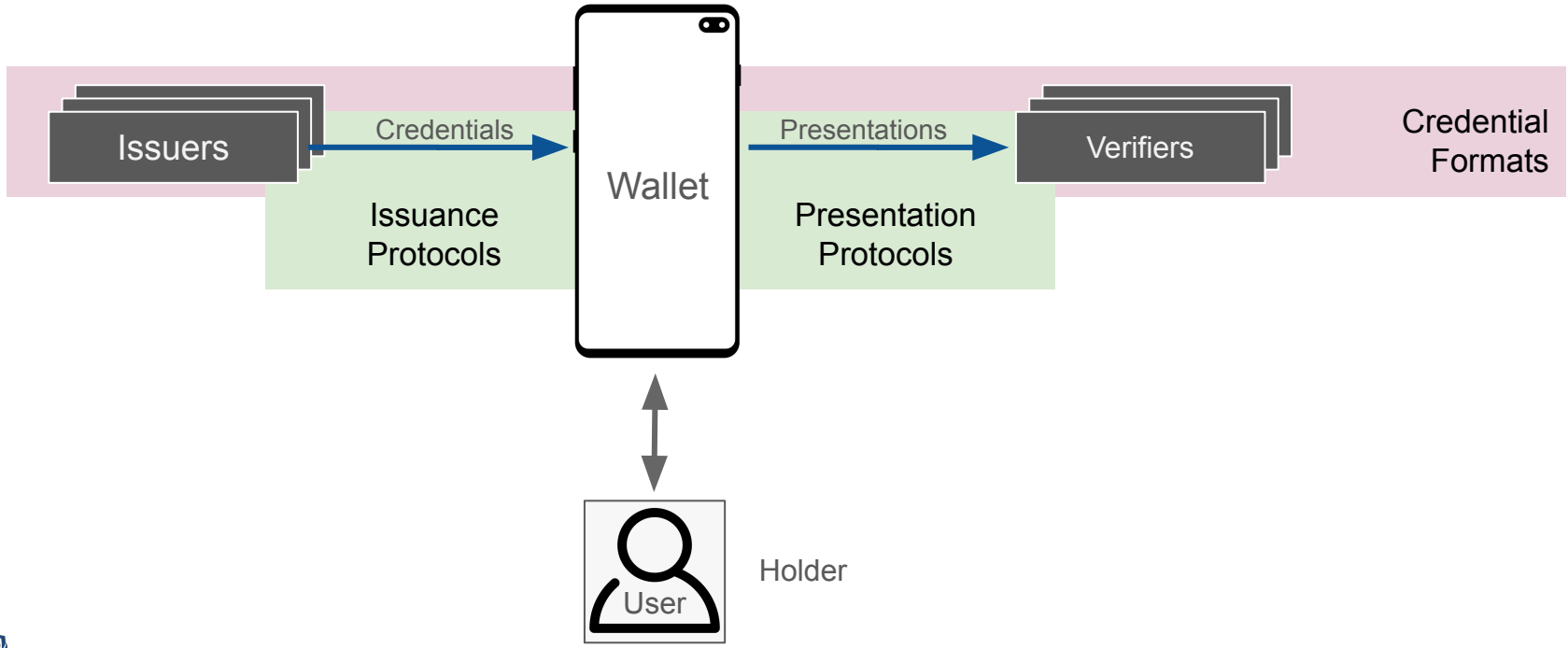
AUTHLETE

Verifiable Credential Ecosystems



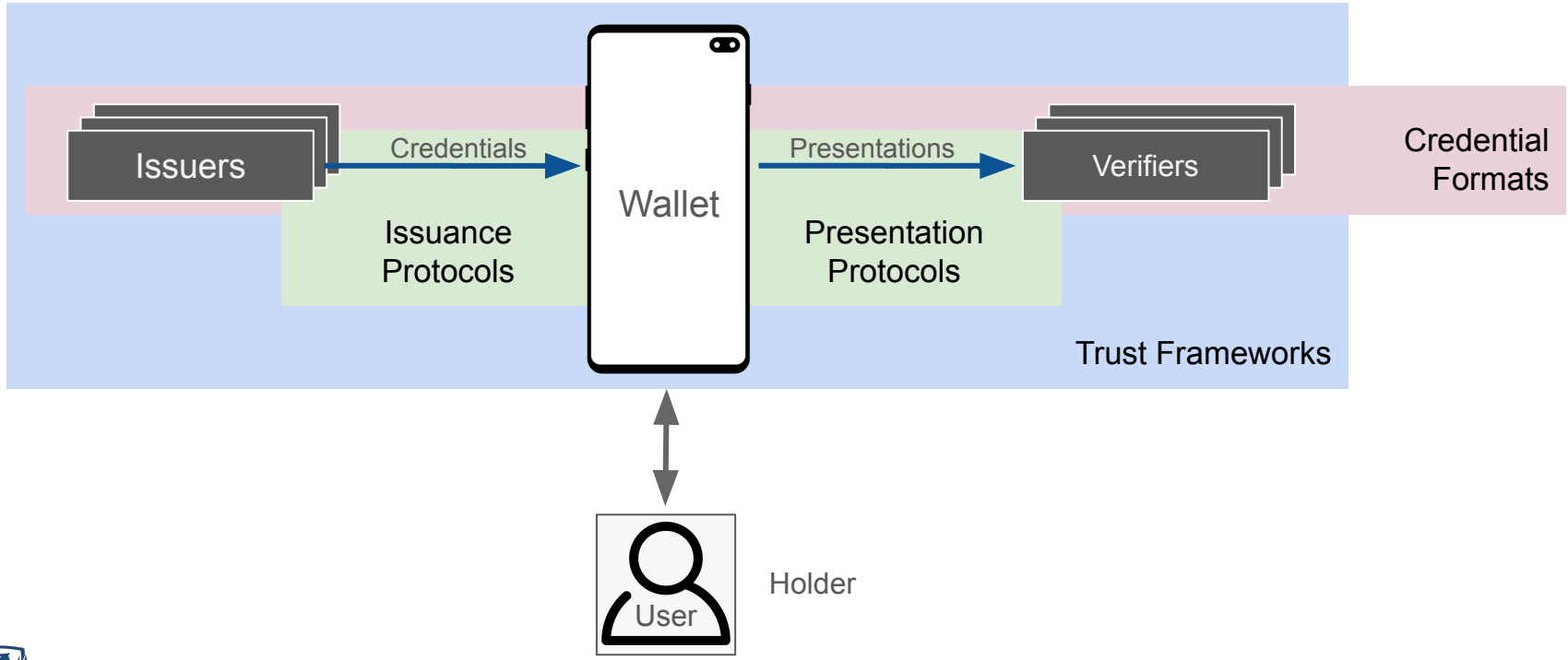
AUTHLETE

Verifiable Credential Ecosystems



AUTHLETE

Verifiable Credential Ecosystems



AUTHLETE

What does Security mean?

Security Properties

Authenticity of Credentials

Credentials can only be issued by authorized Issuers.

Credentials cannot be forged or manipulated by attackers or malicious users.

Key Binding

Key-bound credentials cannot be presented without the holder’s intent.
(Key may be bound to device for copy protection.)

Session Binding

User knows in which context a Credential is presented and to whom.

AUTHLETE

Privacy Properties

Data Minimization

Only the data required for a certain use case is released,
even if credentials contain more claims.

Unlinkability

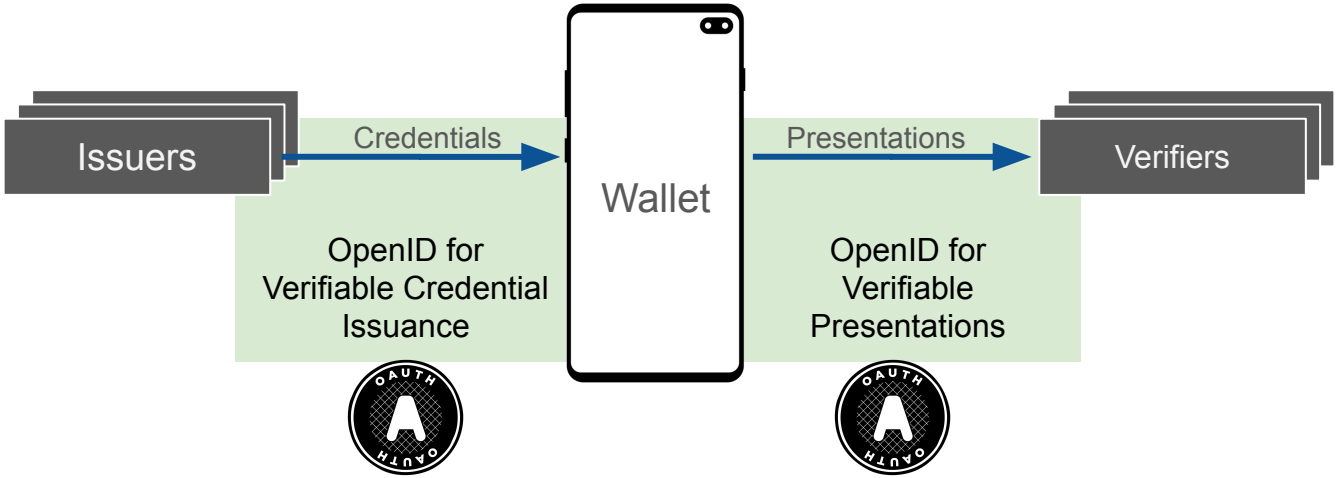
Two presentations cannot be linked together.

Deniability/Repudiation

Verifiers cannot prove the correctness of their data to third parties.

Is this all secure?

Protocols for Verifiable Credential Ecosystems



OpenID for Verifiable Credential Issuance

- Wallet acts as OAuth Client
- Issuer acts as Authorization Server

OpenID for Verifiable Presentations

- Verifier acts as OAuth Client (Relying Party)
- Wallet acts as Authorization Server

Using OAuth is a Good Idea™

- Existing implementations
- Years of practical experience
- Existing “add-on” standards, e.g.,
  - mTLS
  - DPoP
  - PKCE
- Natural companion for JWT (SD-JWT)
- Well-understood security

OAuth has been analyzed extensively

- More than 20 academic research papers
  - Formal models in UC (Universal Composability), ProVerif, Alloy, Web Infrastructure Model
  - Security proofs for OAuth and OpenID Connect
- OAuth Security Best Current Practice RFC (almost finished, I promise!)
- Security proof for combination of issuance and presentation

Great, are we done here?

"Any headline that ends in a question mark can be answered by the word *no*."

AUTHLETE

Ok, what’s left to do?

(Selected) Open Security Topics in Credential Ecosystems

Phishing in Same-Device Flows

Secure Cross-Device Flows

OpenID for Verifiable Presentations

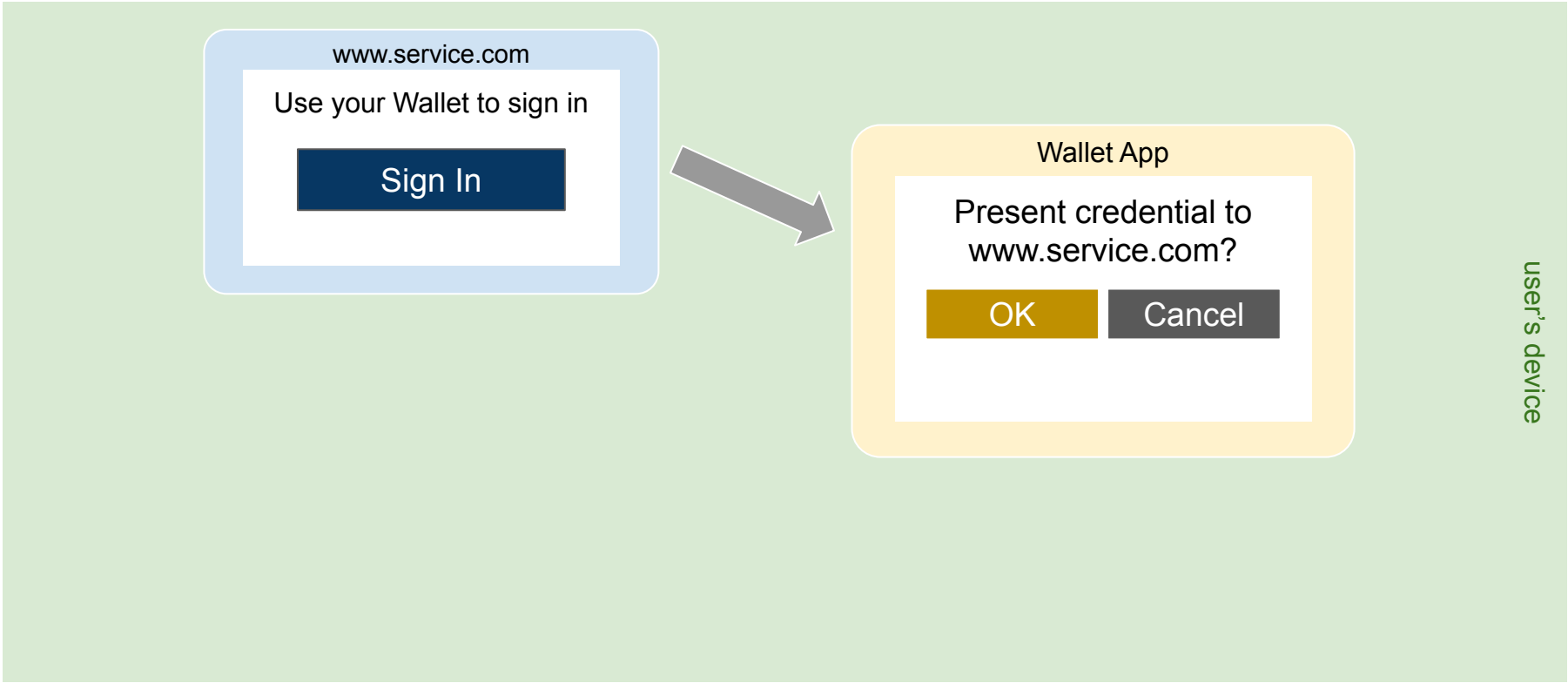
Core principles:

- Verifier acts as OAuth Client
- Wallet acts as Authorization Server
- New vp_token — similar to ID Token — contains presentation
- New response mode direct_post
- New client_id_schema

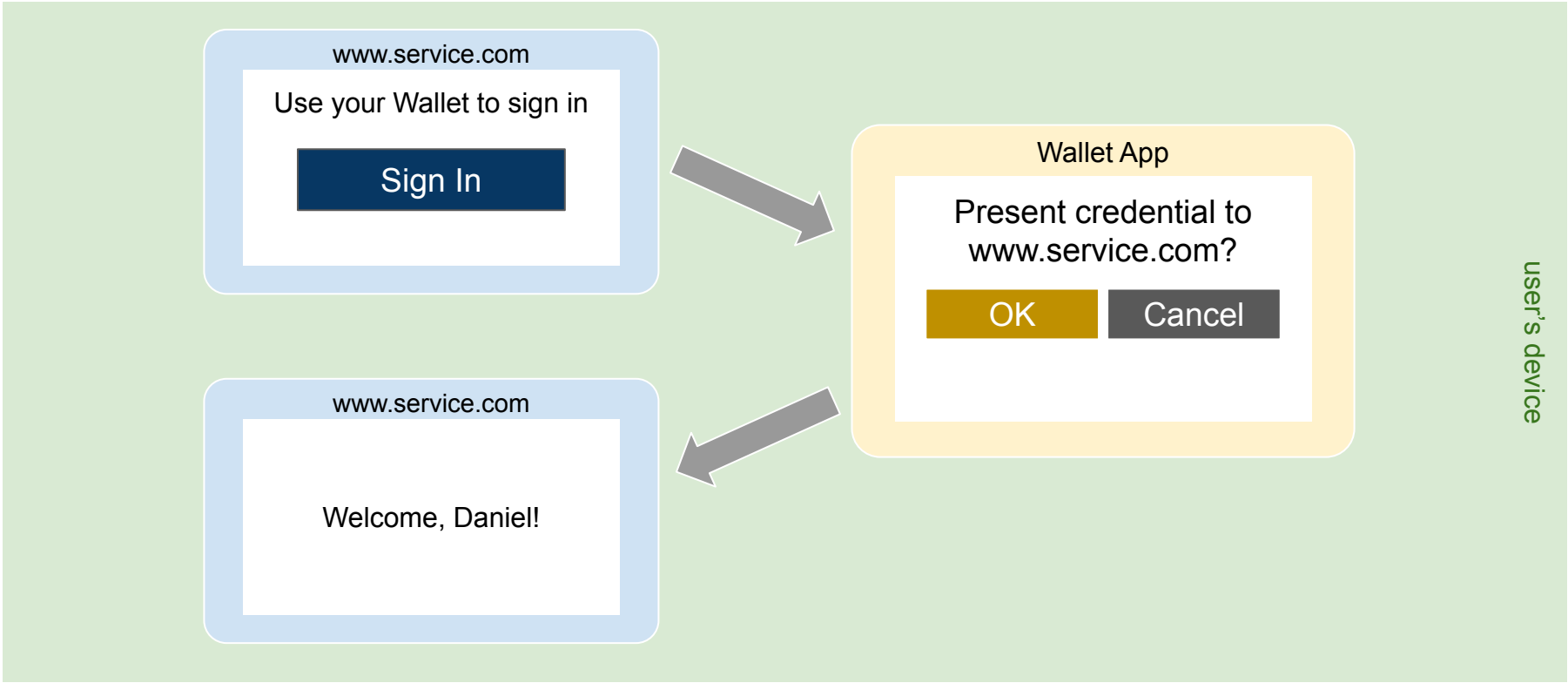
Simple Credential Presentation Flow

user's device

Simple Credential Presentation Flow

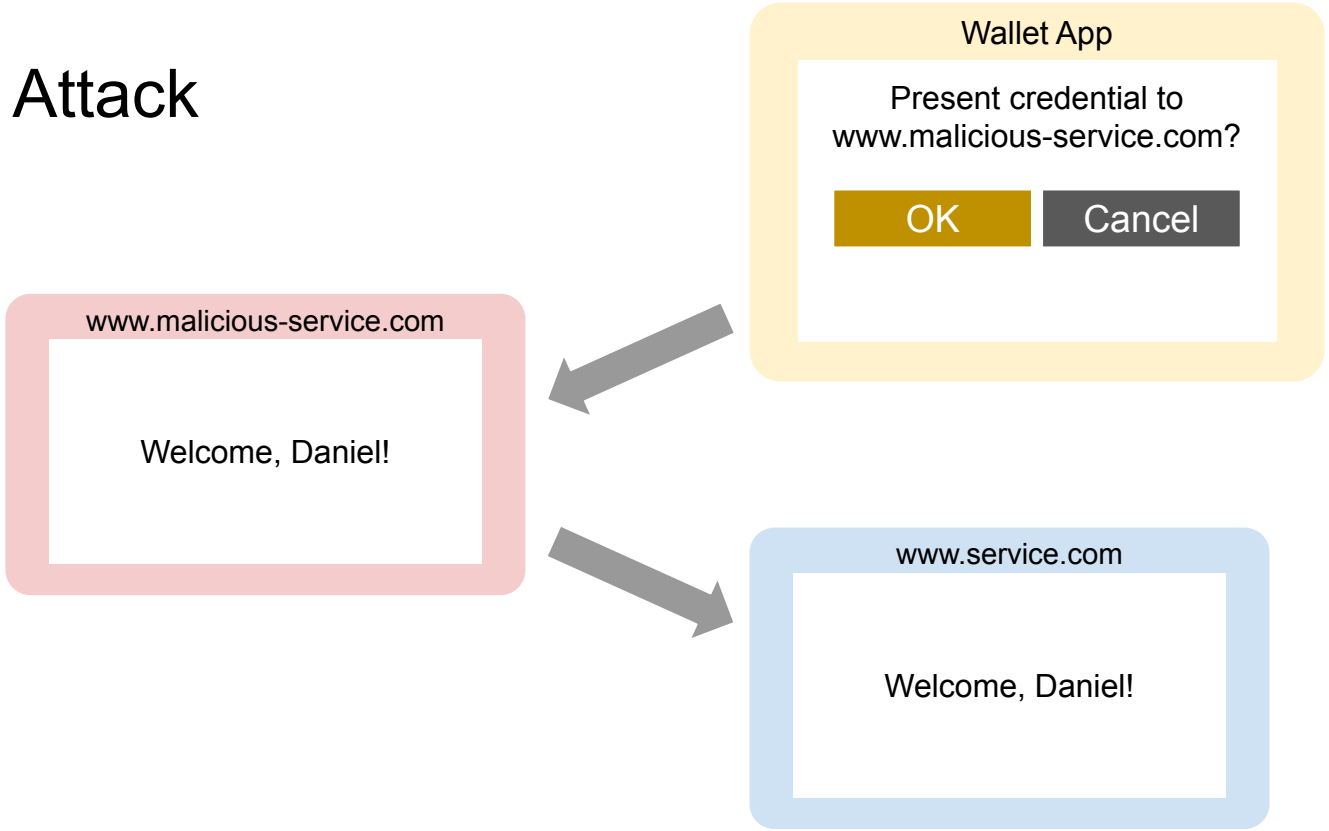


Simple Credential Presentation Flow

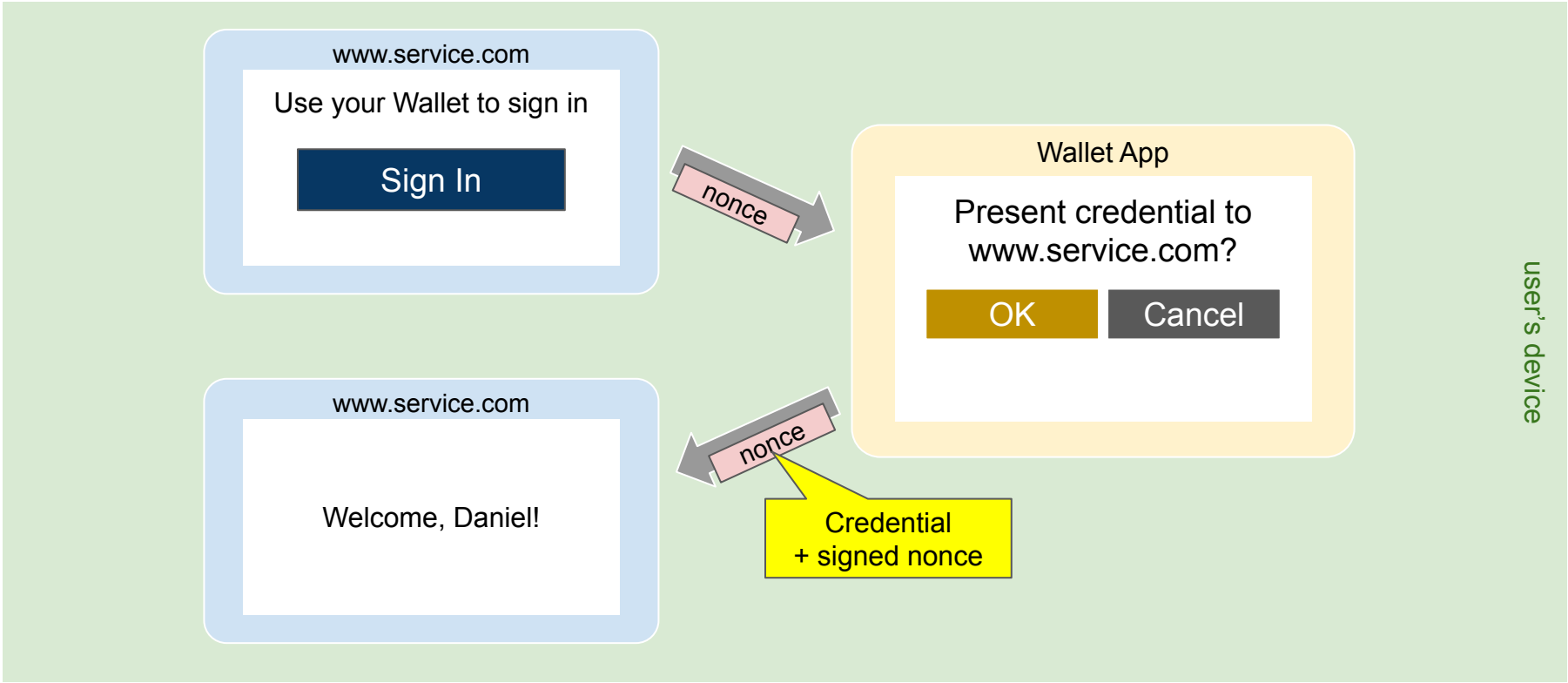


Replay Attacks

Replay Attack



Protection against Replay



Phishing Attacks

Phishing OAuth Flows

attacker's device

www.service.com

Use your Wallet to sign in

Sign In

user's device

Phishing OAuth Flows

attacker's device

www.service.com

Use your Wallet to sign in

Sign In

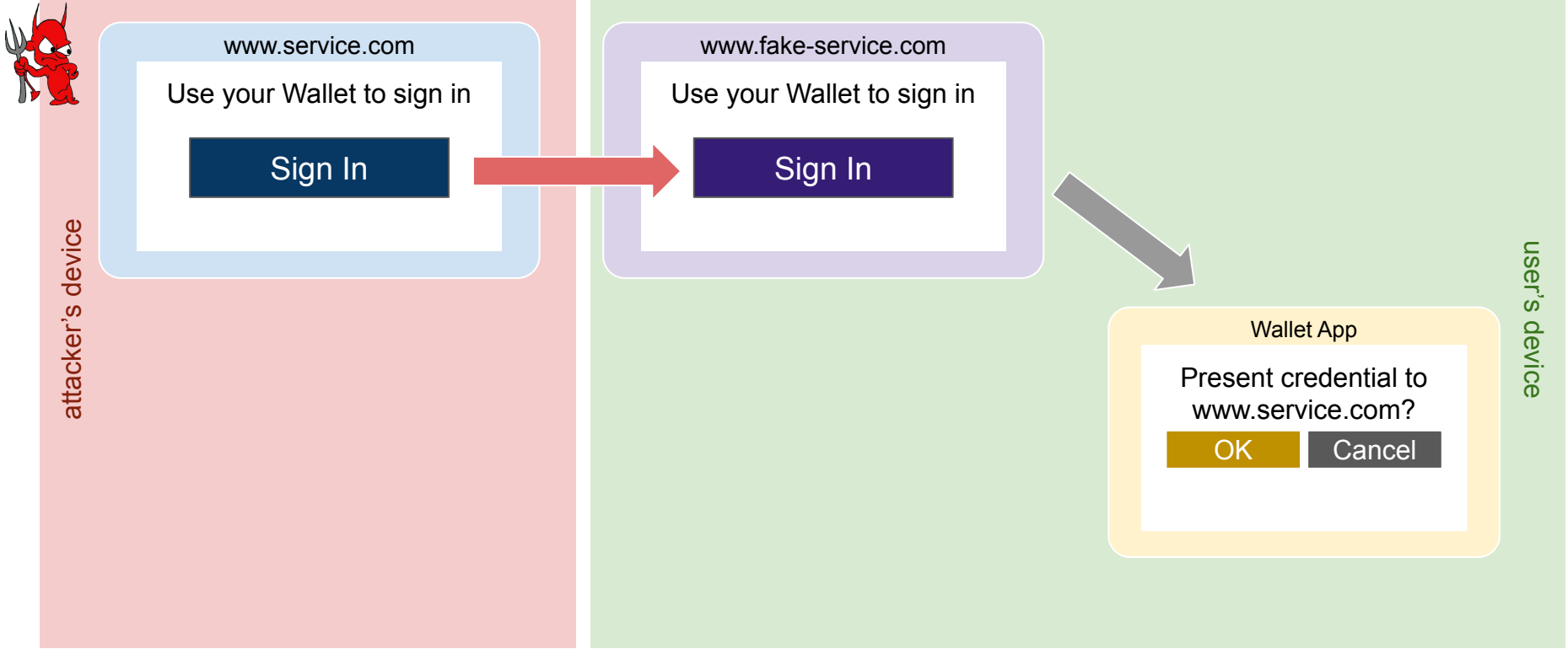
www.fake-service.com

Use your Wallet to sign in

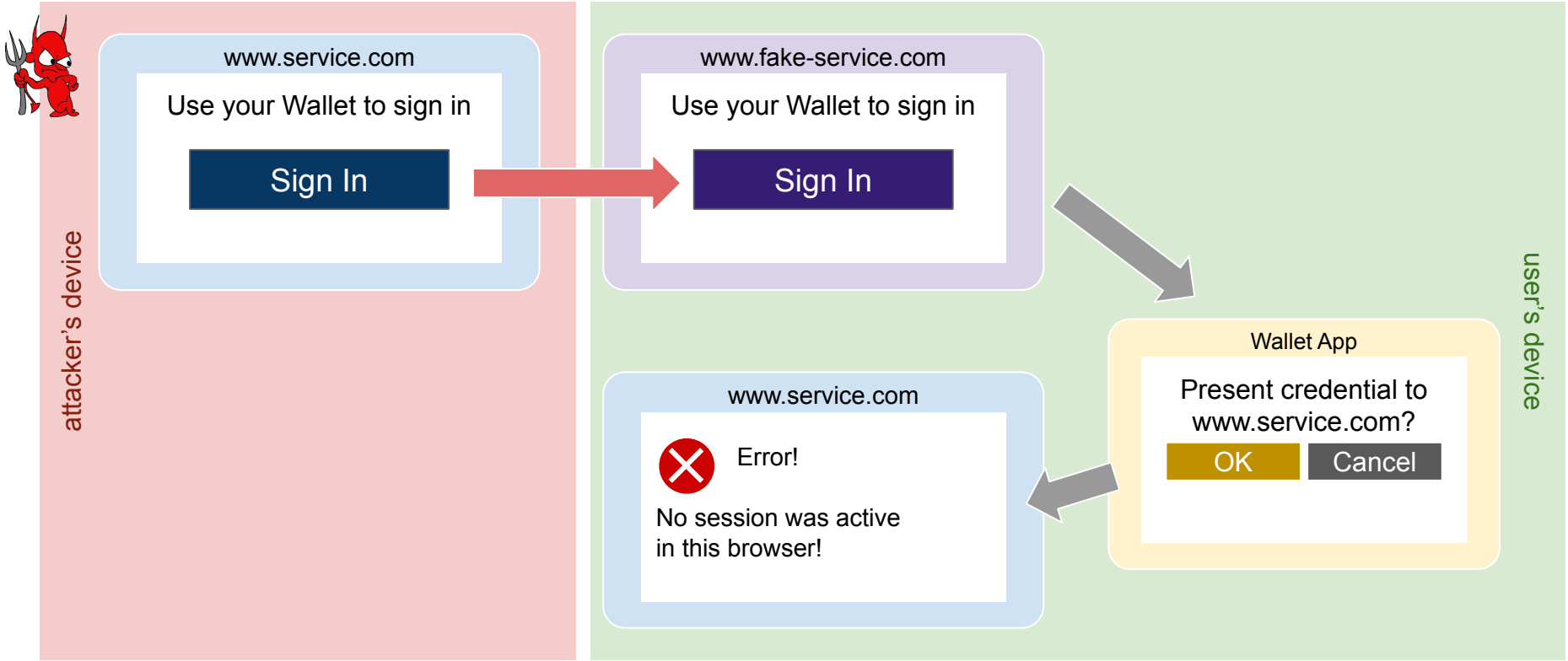
Sign In

user's device

Phishing OAuth Flows



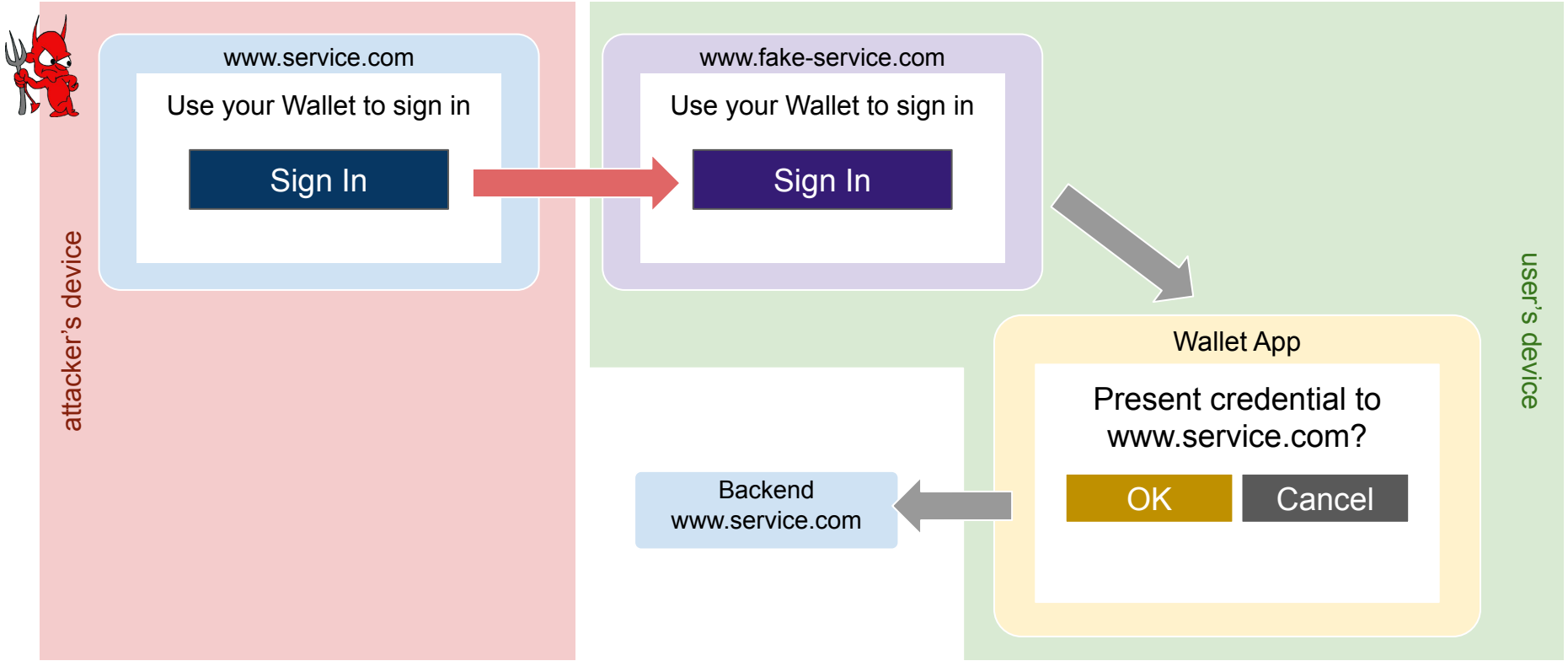
Phishing OAuth Flows



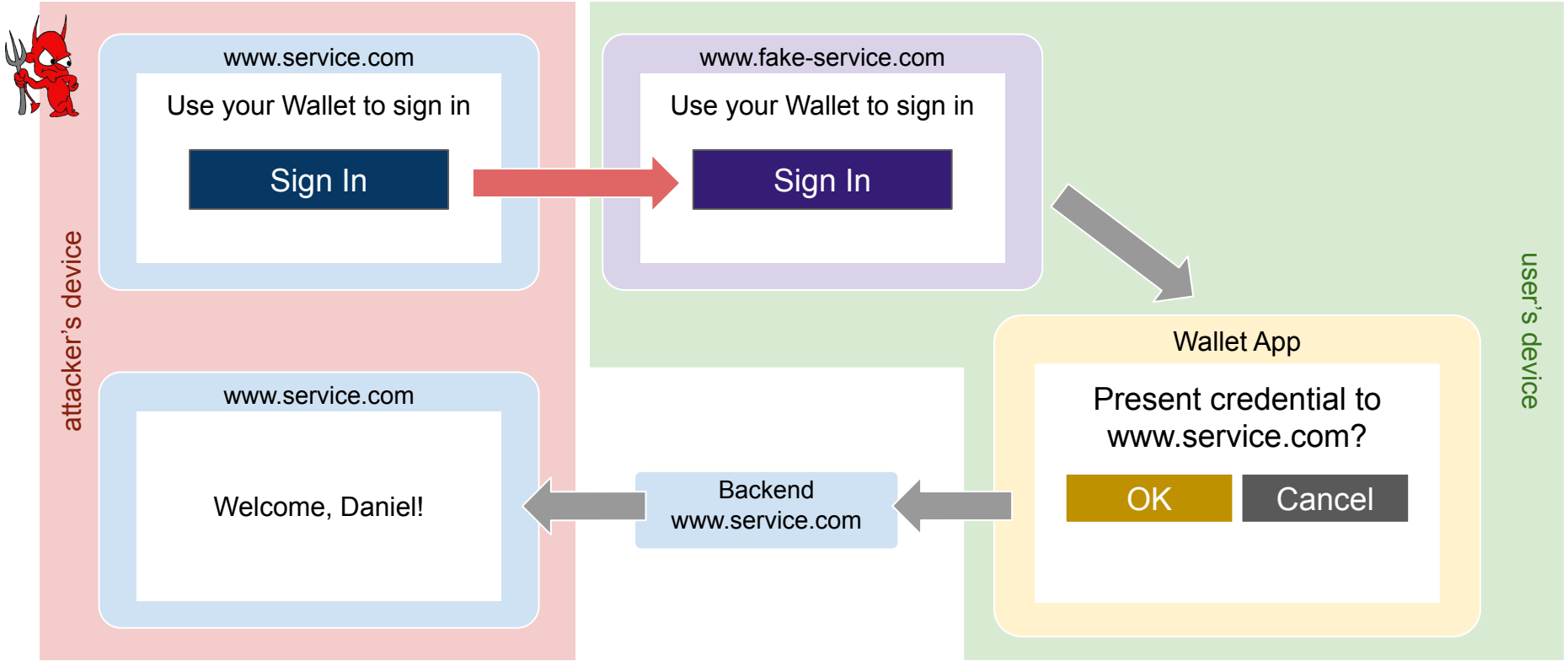
direct_post in OpenID for Verifiable Presentations

- direct_post response mode allows sending the response **directly to the backend of the Verifier**
- Necessary to circumvent the lack of a Wallet backend and size limitations in redirects
- Session in the browser cannot be checked in this case

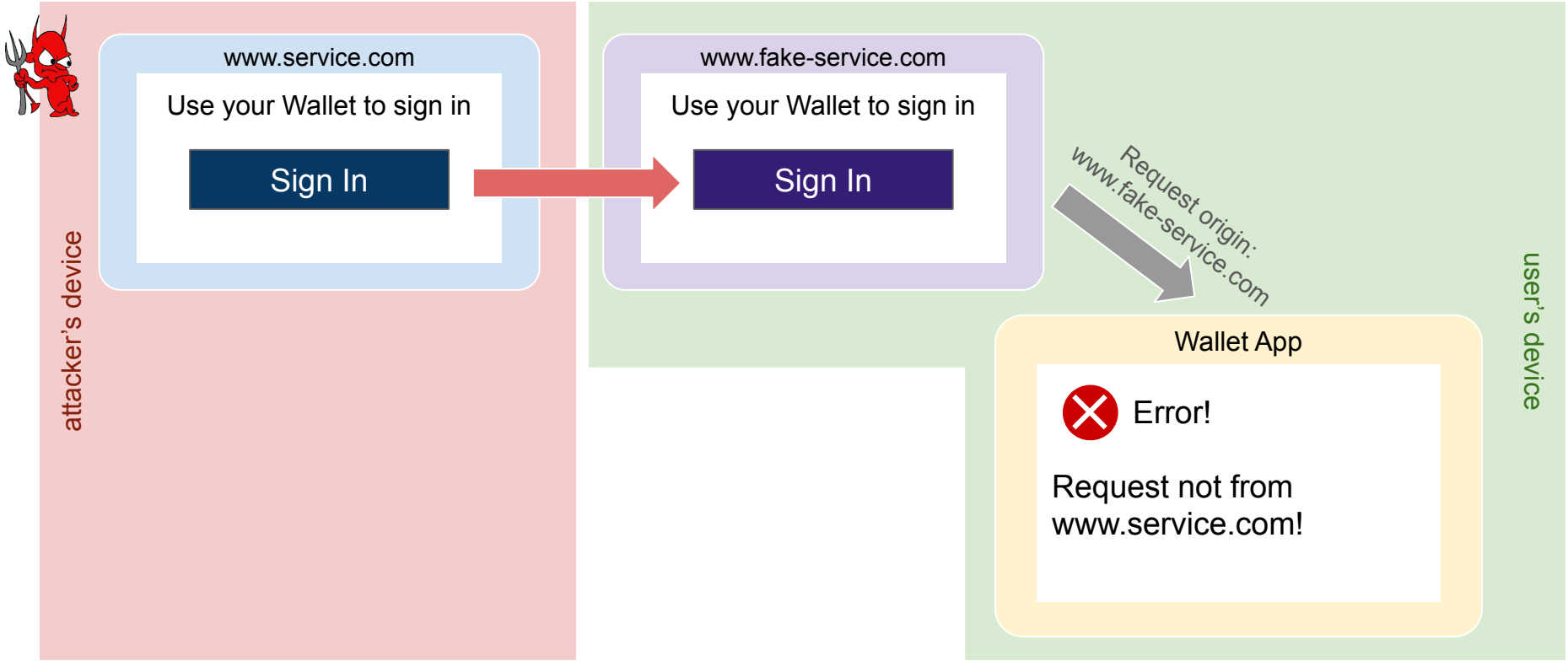
Phishing/Session Binding in Redirect-Based Flows



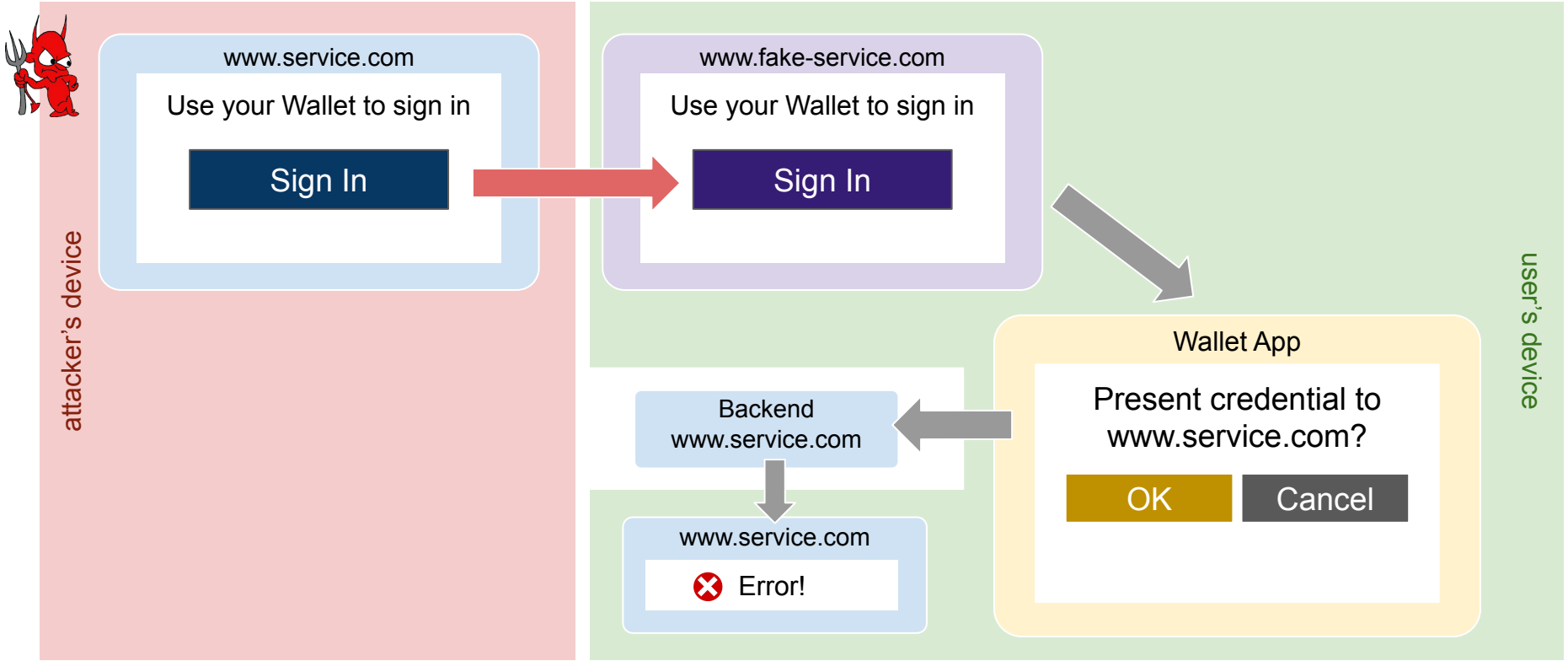
Phishing/Session Binding in Redirect-Based Flows



Solution 1: Trusted Request Origin



Solution 2: Jump back into Frontend



Phishing in Presentation Flows

- Phishing can be a problem for all redirect-based protocols
- Implementations need to protect themselves
- Two solutions in general:
  - Trusted redirect origin — not yet available
  - Redirect back to browser, even if not needed to deliver presentation

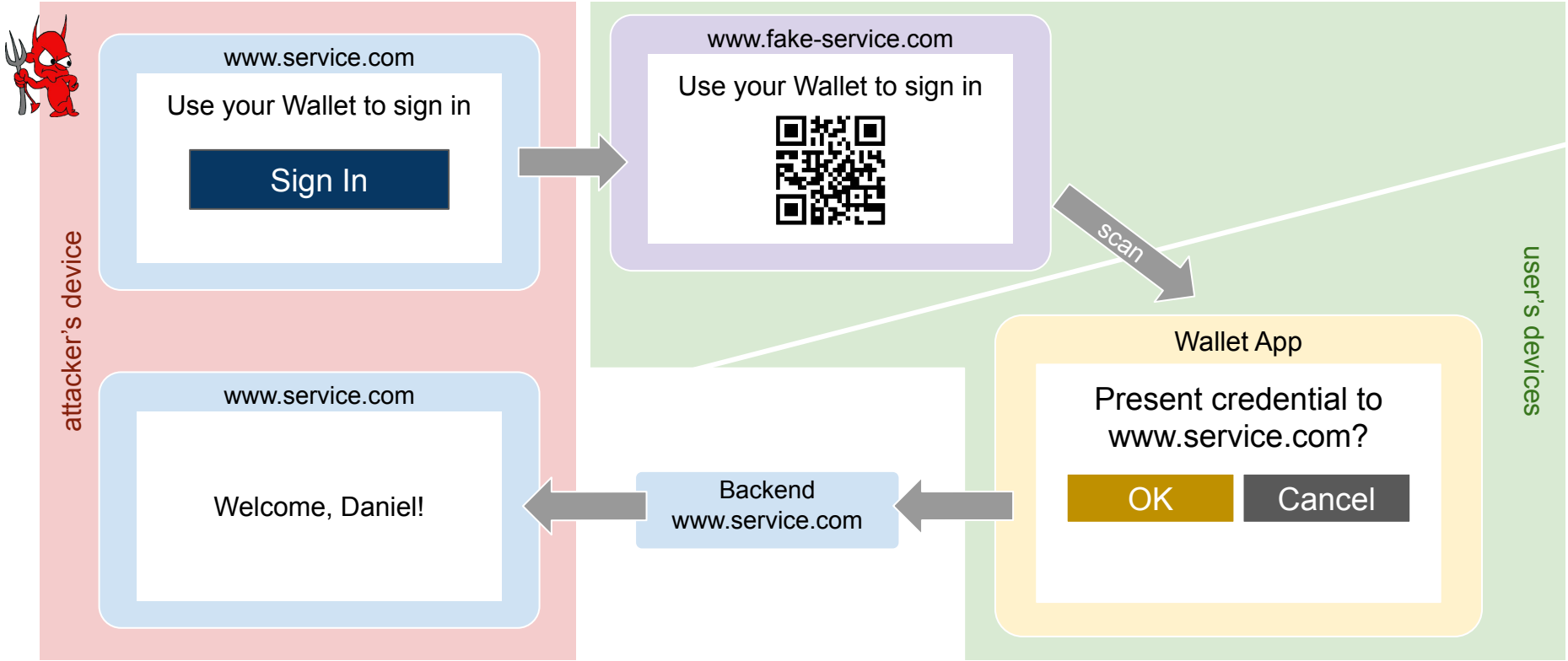
OpenID for Verifiable Presentations:

- Redirect back to browser supported
- Improvement to ‘enforce’ check by Verifier backend under discussion

AUTHLETE

Secure Cross-Device Flows

What about Cross-Device Flows?



Secure Cross-Device Flows

- Very fundamental, generic problem, not limited to OAuth
- There's a draft for that :-)
  draft-ietf-oauth-cross-device-security
- Various approaches to reduce risk:
  - Improve user education
  - Geolocation
  - Network location
  - Heuristics
  - Proximity checks

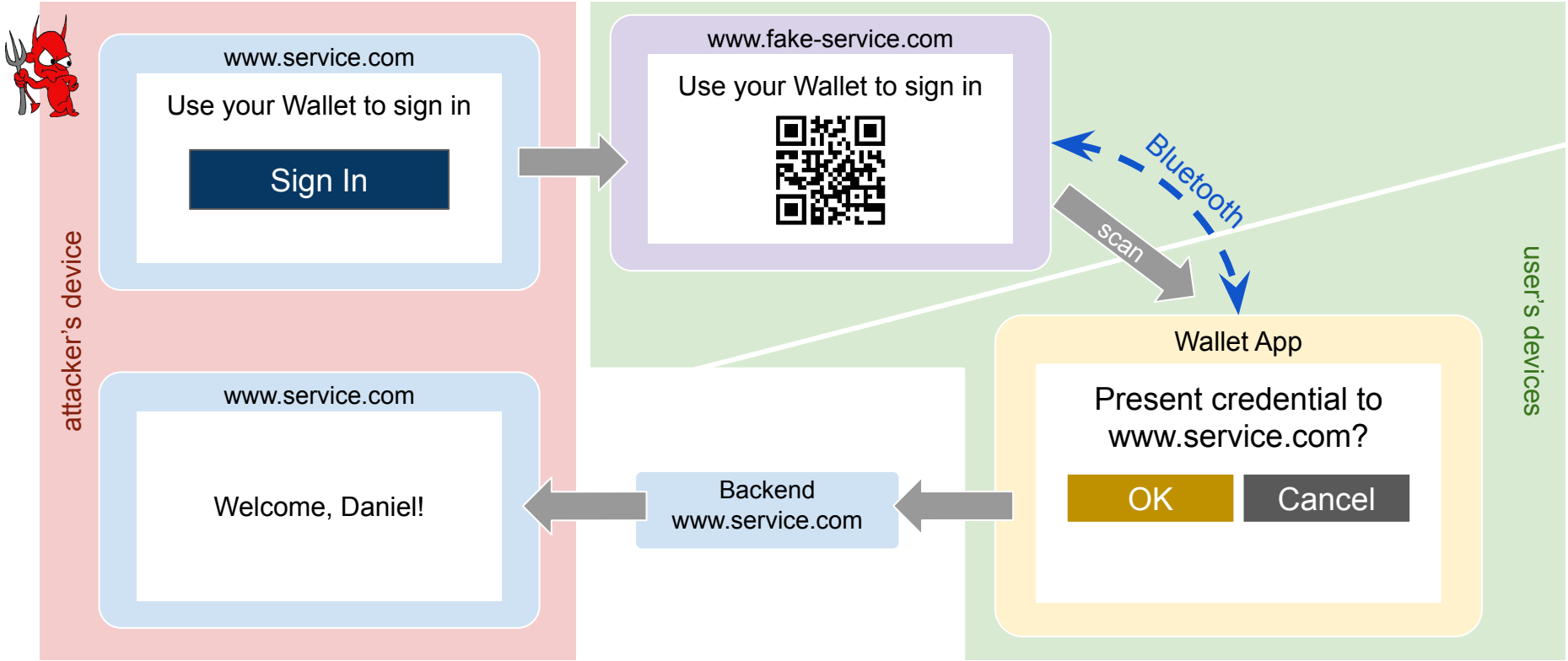
AUTHLETE

Secure Cross-Device Flows

- Very fundamental, generic problem, not limited to OAuth
- There's a draft for that :-)
  draft-ietf-oauth-cross-device-security
- Various approaches to reduce risk:
  - Improve user education
  - Geolocation
  - Network location
  - Heuristics
  - **Proximity checks — e.g. via Bluetooth**

AUTHLETE

Enforce Proximity via Bluetooth



Secure Cross-Device Flows

- State of the Art: No good mechanism for securing such flows (independent of the protocol used!)
- A credential API in the browser/OS could resolve this
- Not yet available or deployed

https://wicg.github.io/digital-identities/

AUTHLETE

Summary

Securing Verifiable Credential Ecosystems

- Reusing what's already defined & tested is great
- Security does not automatically translate to new applications & extensions
- Unsolved problems still exist — but we're working on it

Securing Verifiable Credential Ecosystems

- Reusing what's already defined & tested is great
- Security does not automatically translate to new applications & extensions
- Unsolved problems still exist — but we're working on it

Daniel Fett
Authlete Inc.
daniel.fett@authlete.com
Linkedin:

Thank you!