Authentication Services / ASCredentialUpdater Deprecated

Class

ASCredentialUpdater Deprecated

A class to pass credential update events to credential managers enabled on the system.

iOS 26.0–26.2 Deprecated | iPadOS 26.0–26.2 Deprecated | Mac Catalyst 26.0–26.2 Deprecated | macOS 26.0–26.2 Deprecated | visionOS 26.0–26.2 Deprecated

```
final class ASCredentialUpdater
```

Deprecated

Use `ASCredentialDataManager` instead

Overview

The `ASCredentialUpdater` implements the functionality of the WebAuthn Signal API, allowing apps to update credential managers with information about existing credentials. By informing credential managers of updated, removed, or revoked credentials, the credential managers can stay synchronized with the credential information of the person using the device.

Use `ASCredendialUpdater` in the following scenarios:

Updating passkey metadata

The system UI represents passkeys by a “user name”, set when the passkey is created. This field is usually a user name or email address. If your app allows the person using it to change the user name for their account, use `reportPublicKeyCredentialUpdate(relyingPartyIdentifier:userHandle:newName:)` to relay that information to credential managers.

Removing revoked passkeys

Your app and its related services may allow someone using it to remove passkeys associated with their account, or delete the account entirely. Use reportUnknownPublicKeyCredential(relyingPartyIdentifier:credentialID:) to inform credential managers of this deletion, so the passkeys aren’t shown in future login flows.

Removing passwords

When you’re confident the person using your app has transitioned to using a passkey to sign in and no longer requires a password fallback to sign in, you can direct credential managers to remove or hide passwords with reportUnusedPasswordCredential(domain:userName:). This prevents the now invalid passwords from appearing in the UI.

The following example shows how an app might use this class when processing various sign-in and account-management events:

```
import AuthenticationServices

let credentialUpdater = ASCredentialUpdater()

func handleSuccessfulPasskeySignIn() {
    ...
    // Update passkey if the username changed on the account.
    try credentialUpdater.reportPublicKeyCredentialUpdate(relyingPartyIdentifier:"ex

    // Report accepted credentials.
    try credentialUpdater.reportAllAcceptedPublicKeyCredentials(relyingPartyIdentifi

    // Remove or hide stale password.
    try credentialUpdater.reportUnusedPasswordCredential(domain: "example.com", user
}

func handleFailedPasskeySignIn() {
    ...
    // Remove or hide invalid passkey.
    try credentialUpdater.reportUnknownPublicKeyCredential(relyingPartyIdentifier:"e
}

func handleAccountAccountDeletion() {
    ...
    try credentialUpdater.reportAllAcceptedPublicKeyCredentials(relyingPartyIdentifi
    try credentialUpdater.reportUnusedPasswordCredential(domain: "example.com", user
}
```

> Note
>
> To protect the privacy of the person using the app, this class’s methods don’t indicate whether their operations succeeded. A successful call only indicates that the parameters were well formed.

Topics

Creating a credential updater

~~init()~~

Creates an instance of the credential updater class.

Reporting accepted credentials

~~func reportAllAcceptedPublicKeyCredentials(relyingPartyIdentifier:~~
~~String, userHandle: Data, acceptedCredentialIDs: [Data]) async throws~~

Provides credential managers with a snapshot of all credential identifiers accepted for a given user handle.

Reporting updated credentials

~~func reportPublicKeyCredentialUpdate(relyingPartyIdentifier: String,~~
~~userHandle: Data, newName: String) async throws~~

Provides credential managers with an update to a credential’s name, such as when changing the user name or email address on an account.

Reporting unused and unknown credentials

~~func reportUnusedPasswordCredential(domain: String, userName: String)~~
~~async throws~~

Informs credential managers that a password is no longer in use.

~~func reportUnknownPublicKeyCredential(relyingPartyIdentifier: String,~~
~~credentialID: Data) async throws~~

Informs credential managers that a specific credential is unknown or no longer accepted.