Framework

Authentication Services

Make it easy for users to log into apps and services.

iOS 12.0+ | iPadOS 12.0+ | Mac Catalyst 13.0+ | macOS 10.15+ | tvOS 13.0+ | visionOS 1.0+ | watchOS 6.0+

Overview

Use the Authentication Services framework to improve the experience of users when they enter credentials to establish their identity.

- Give users the ability to sign into your services with their Apple ID.
- Enable users to look up their stored passwords from within the sign-in flow of an app.
- Provide a passwordless registration and authentication workflow for apps and websites using iCloud Keychain or a physical security key.
- Perform automatic security upgrades from weak to strong passwords, or upgrade to using Sign in with Apple.
- Share data between an app and a web browser using technologies like OAuth to leverage existing web-based logins in the app.
- Create a single sign-on (SSO) experience in an enterprise app.

Simple and straightforward sign-up and sign-in flows reduce the burden on the user to remember passwords, which may improve security.

Topics

Authorization requests

```
class ASAuthorizationController
```

A controller that manages authorization requests that a provider creates.

`struct AuthorizationController`

A SwiftUI environment value that views use to perform authorization requests.

`enum ASAuthorizationResult`

Describes the outcome of a successful authorization request.

Sign In with Apple

{} Implementing User Authentication with Sign in with Apple

Provide a way for users of your app to set up an account and start using your services.

{} Simplifying User Authentication in a tvOS App

Build a fluid sign-in experience for your tvOS apps using AuthenticationServices.

`struct SignInWithAppleButton`

A SwiftUI view that creates the Sign in with Apple button for display.

`Sign in with Apple Entitlement`

An entitlement that lets your app use Sign in with Apple.

`class ASAuthorizationAppleIDProvider`

A mechanism for generating requests to authenticate users based on their Apple ID.

`class ASAuthorizationAppleIDCredential`

A credential that results from a successful Apple ID authentication.

Passwords

🗎 Password AutoFill

Streamline your app’s login and onboarding procedures.

`class ASAuthorizationPasswordProvider`

A mechanism for generating requests to perform keychain credential sharing.

`class ASPasswordCredential`

A password credential.

☰ Password use in web browsers

Register and authenticate website users by using passwords.

Passkeys

Public-Private Key Authentication

Register and authenticate users with passkeys and security keys, without using passwords.

Passkey use in web browsers

Register and authenticate website users by using passkeys.

Performing fast account creation with passkeys

Allow people to quickly create an account with passkeys and associated domains.

Connecting to a service with passkeys

Allow users to sign in to a service without typing a password.

Web authentication sessions

Authenticating a User Through a Web Service

Use a web authentication session to authenticate a user in your app.

Securing Logins with iCloud Keychain Verification Codes

Use time-based codes generated on-device for a secure authentication experience.

`class ASWebAuthenticationSession`

A session that an app uses to authenticate a user through a web service.

`struct WebAuthenticationSession`

A SwiftUI environment value that views use to authenticate someone using a web service.

Supporting Single Sign-On in a Web Browser App

Extend your web browser app to handle web authentication requests from other apps.

`class ASWebAuthenticationSessionWebBrowserSessionManager`

A session manager that mediates sharing data between an app and a web browser.

`ASWebAuthenticationSessionWebBrowserSupportCapabilities`

A collection of keys that a browser app uses to declare its ability to handle authentication requests from other apps.

AutoFill credentials

Providing one-time passcodes to AutoFill

Help people efficiently perform multifactor authentication.

`AutoFill Credential Provider Entitlement`

A Boolean value that indicates whether the app may, with user permission, provide user names and passwords for AutoFill in Safari and other apps.

`class ASCredentialProviderViewController`

A view controller that a credential manager app uses to extend AutoFill.

Credential migration

`class ASCredentialExportManager`

A class to manage exporting credentials.

`class ASCredentialImportManager`

A class to manage importing credentials.

Single sign-on (SSO)

Enterprise single sign-on (SSO)

Platform single sign-on (SSO)

Use credentials from macOS login to perform single sign-on with an identity provider.

Apple TV authentication

`var customAuthorizationMethods: [ASAuthorizationCustomMethod]`

An array of custom authorization methods for the user to choose.

`func authorizationController(ASAuthorizationController, didCompleteWithCustomMethod: ASAuthorizationCustomMethod)`

Informs the delegate when authorization completes, and specifies the custom method the user selected.

`struct ASAuthorizationCustomMethod`

The custom authorization method.

Automatic security upgrades

Upgrading Account Security With an Account Authentication Modification Extension

Automatically and transparently convert accounts to Sign in with Apple or to use strong passwords for improved security.

`class ASAccountAuthenticationModificationController`

An object that performs a request to modify an account's authentication properties.

`class ASAccountAuthenticationModificationViewController`

A view controller that can upgrade user passwords to strong passwords, or convert accounts to use Sign in with Apple.

`class ASAccountAuthenticationModificationExtensionContext`

An object that you interact with to change an account's password or to upgrade to Sign in with Apple.

Updating credential managers

~~`class ASCredentialUpdater`~~

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

Deprecated

Reference

AuthenticationServices Enumerations

AuthenticationServices Data Types

Classes

`class ASAuthorizationAccountCreationPlatformPublicKeyCredential`

`class ASAuthorizationAccountCreationPlatformPublicKeyCredentialRequest`

`class ASAuthorizationAccountCreationProvider`

`class ASAuthorizationProviderExtensionUserLoginConfiguration`

`class ASCredentialDataManager`

This class allows submitting credentials and events to any credential manager enabled on the system.

Beta

`class ASGeneratePasswordsRequest` Beta

`class ASGeneratedPassword` Beta

`class ASOneTimeCodeCredentialIdentity`

`class ASSavePasswordRequest` Beta

Structures

`struct ASAuthorizationProviderExtensionEncryptionAlgorithm`

`struct ASAuthorizationProviderExtensionSigningAlgorithm`

`struct ASAutoFillURLScope`

This structure represents the subset of URL components supported for the AutoFill of credentials.

Beta

`struct ASEmailIdentifier`

`struct ASImportableCredentialScope`

The scope for where a credential should be usable.

`struct ASImportableEditableField`

A field that someone can edit within a credential.

`struct ASPhoneNumberIdentifier`

`struct ASPublicKeyCredentialClientData`

`struct CredentialDataManager` Beta

Variables

`let ASCredentialExchangeActivity: String`

The activity type used in user activity objects sent to importing apps.

`let ASCredentialImportToken: String`

The key for the token in the user info dictionary of the user activity sent to importing apps.

Enumerations

`enum ASContactIdentifier`

```
enum ASContactIdentifierRequest
```