

Xcode / Projects and workspaces / Allowing apps and websites to link to your con... / Supporting associated domains

Supporting associated domains

Connect your app and a website to provide both a native app and a browser experience.

Overview

Associated domains establish a secure association between domains and your app so you can share credentials or provide features in your app from your website. For example, an online retailer may offer an app to accompany their website and enhance the user experience.

Shared web credentials, universal links, Handoff, and App Clips all use associated domains. Associated domains provide the underpinning to universal links, a feature that allows an app to present content in place of all or part of its website. Users who don't download the app get the same information in a web browser instead of the native app.

To associate a website with your app, you need to have the associated domain file on your website and the appropriate entitlement in your app. The apps in the apple-app-site-association file on your website must have a matching [Associated Domains Entitlement](#).

Add the associated domain file to your website

When a user installs your app, the system attempts to download the associated domain file and verify the domains in your entitlement.

Note

If your site uses multiple subdomains (such as example.com, www.example.com, and support.example.com), each requires its own entry in the [Associated Domains Entitlement](#), and each must serve its own apple-app-site-association file.

To add the associated domain file to your website, create a file named apple-app-site-association (without an extension). Update the JSON code in the file for the services you support on the domain. For universal links, be sure to list the app identifiers for your domain in the applinks service. Similarly, if you create an App Clip, be sure to list your App Clip's app identifier using the appclips service.

The following JSON code represents the contents of a simple association file:

```
{
  "applinks": {
    "details": [
      {
        "appIDs": [ "ABCDE12345.com.example.app", "ABCDE12345.com.example.app2" ],
        "components": [
          {
            "##": "no_universal_links",
            "exclude": true,
            "comment": "Matches any URL with a fragment that equals no_universal_links and instr
          },
          {
            "/": "/buy/*",
            "comment": "Matches any URL with a path that starts with /buy/."
          },
          {
            "/": "/help/website/*",
            "exclude": true,
            "comment": "Matches any URL with a path that starts with /help/website/ and instruct
          },
          {
            "/": "/help/*",
            "?": { "articleNumber": "?????" },
            "comment": "Matches any URL with a path that starts with /help/ and that has a query
          }
        ]
      }
    ],
    "webcredentials": {
      "apps": [ "ABCDE12345.com.example.app" ]
    },
    "appclips": {
      "apps": [ "ABCDE12345.com.example.MyApp.Clip" ]
    }
  }
}
```

The appIDs and apps keys specify the application identifiers for the apps that are available for use on this website along with their service types. Use the following format for the values in these keys:

<Application Identifier Prefix>.<Bundle Identifier>

The details dictionary only applies to the [applinks](#) service type; other service types don't use it. The components key is an array of dictionaries that provides pattern matching for components of the URL.

After you construct the association file, place it in your site's .well-known directory. The file's URL should match the following format:

https://<fully qualified domain>/.well-known/apple-app-site-association

You must host the file using https:// with a valid certificate and with no redirects.

Add the associated domains entitlement to your app

To set up the entitlement in your app, open the target's Signing & Capabilities tab in Xcode and add the Associated Domains capability. If they're not already present, this step adds the [Associated Domains Entitlement](#) to your app and the associated domains feature to your app ID.

Important

For a single-target watchOS apps, add the Associated Domains capability to the watchOS app target. For watchOS apps with separate WatchKit extensions, you must add the Associated Domains capability to the WatchKit Extension target.

To add your domain to the entitlement, click Add (+) at the bottom of the Domains table to add a placeholder domain. Replace the placeholder with the appropriate prefix for the service your app supports and your site's domain. Make sure to only include the desired subdomain and the top-level domain. Don't include path and query components or a trailing slash (/).

Add the domains that share credentials with your app. For services other than appclips, you can prefix a domain with *. to match all of its subdomains.

Each domain you specify uses the following format:

<service><fully qualified domain>

Starting with macOS 11 and iOS 14, apps no longer send requests for apple-app-site-association files directly to your web server. Instead, they send these requests to an Apple-managed content delivery network (CDN) dedicated to associated domains.

While you're developing your app, if your web server is unreachable from the public internet, you can use the alternate mode feature to bypass the CDN and connect directly to your private domain.

You enable an alternate mode by adding a query string to your associated domain's entitlement as follows:

<service><fully qualified domain>?mode=<alternate mode>

For more information about alternate modes, see [Associated Domains Entitlement](#). For information about universal links, see [Allowing apps and websites to link to your content](#). For information about Handoff, see [Implementing Handoff in Your App](#). For information about App Clips, see [Configuring App Clip experiences](#).

Important

Apple's content delivery network requests the apple-app-site-association file for your domain within 24 hours. Devices check for updates approximately once per week after app installation.

Topics

Entitlements

[Associated Domains Entitlement](#)

The associated domains for specific services, such as shared web credentials, universal links, and App Clips.

Services

object [applinks](#)

The root object for a universal links service definition.

Platforms

iOS
iPadOS
macOS
tvOS
visionOS
watchOS
Tools
Swift
SwiftUI
Swift Playground
TestFlight
Xcode
Xcode Cloud
SF Symbols

Topics & Technologies

Accessibility
Accessories
App Extension
App Store
Audio & Video
Augmented Reality
Design
Distribution
Education
Fonts
Games
Health & Fitness
In-App Purchase
Localization
Maps & Location
Machine Learning & AI
Open Source
Security
Safari & Web

Resources

Documentation
Tutorials
Downloads
Forums
Videos
Support
Support Articles
Contact Us
Bug Reporting
System Status
Account
Apple Developer
App Store Connect
Certificates, IDs, & Profiles
Feedback Assistant

Programs

Apple Developer Program
Apple Developer Enterprise Program
App Store Small Business Program
MFI Program
News Partner Program
Video Partner Program
Security Bounty Program
Security Research Device Program
Events
Meet with Apple
Apple Developer Centers
App Store Awards
Apple Design Awards
Apple Developer Academies
WWDC