

Fine-tune your App Transport Security settings

June 16, 2021



At Apple, we believe privacy is a fundamental human right. When people connect to a public Wi-Fi hotspot, they expect to use your app to send and receive data without worrying that someone in the vicinity could intercept their connection and gain access to unencrypted data. Allowing even seemingly-innocuous data to remain unencrypted can expose people to snooping and fingerprinting by anyone on the network.

Transport Layer Security (TLS) uses encryption to protect connections from prying eyes, and URLSession provides strong TLS connections by default with App Transport Security (ATS).

If you need to connect to older servers that don't support TLS, however, you can now add ATS exceptions to your app. Ideally, exceptions should just carve out the specific domains or frameworks that make insecure connections, and you should limit any exceptions you do request. Avoid sending data unencrypted except when absolutely necessary for your app to function.

Identify necessary ATS exceptions

To make sure your app — and the data used within it — is as secure as possible, it's important to identify whether your app is currently making insecure connections.

To check, disable all your active ATS exceptions by setting their values in your Info.plist to "NO." From there, open your app or run your unit tests. If your app makes an insecure connection, Xcode will generate runtime errors for each one.

▼ App Transport Security Settings	↕	Dictionary	(2 items)
Allows Arbitrary Loads for Media	↕	Boolean	NO
▼ Exception Domains	↕	Dictionary	(1 item)
▼ example.com		Dictionary	(1 item)
NSEnvironmentAllowsInsecureHTTPLoads		Boolean	0

If your app is generating insecure connections, there are a few steps you can take to remove them.

Secure your servers

If your app connects to servers you control, make sure those servers support secure connections. This requires a TLS certificate. If you use a hosting service, check whether they offer certificates, and make sure those certificates meet the requirements detailed in “Preventing Insecure Network Connections.”

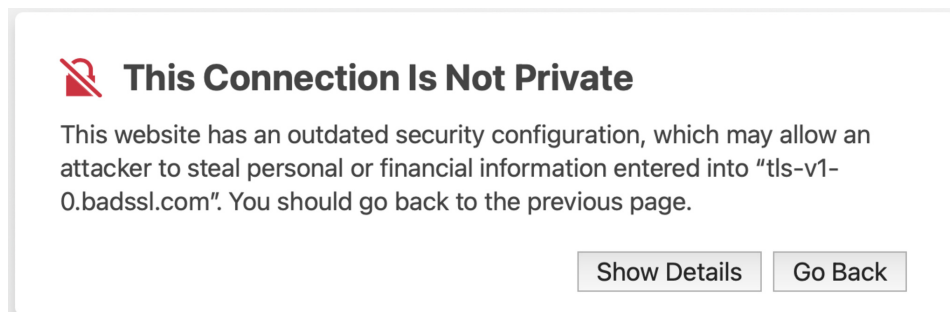
[Preventing Insecure Network Connections >](#)

Use HTTPS

If your app connects to servers you don’t control, you should always attempt to connect to those servers over HTTPS instead of HTTP. You can identify whether a server supports HTTPS by simply changing “http://” to “https://” in your URL string and trying to load data from that website. You can check this manually in a browser, or run code as follows:

```
let request = URLRequest(url: URL(string: "https://www.example.com")!)
```

Many websites redirect HTTP connections to HTTPS. Connecting over HTTPS first can often improve the performance of your app. Note, however, that while a website may use HTTPS, that doesn’t mean it’s ATS-compatible. For instance, it may be using an outdated version of TLS, which, on Safari, displays a “This Connection Is Not Private” warning.

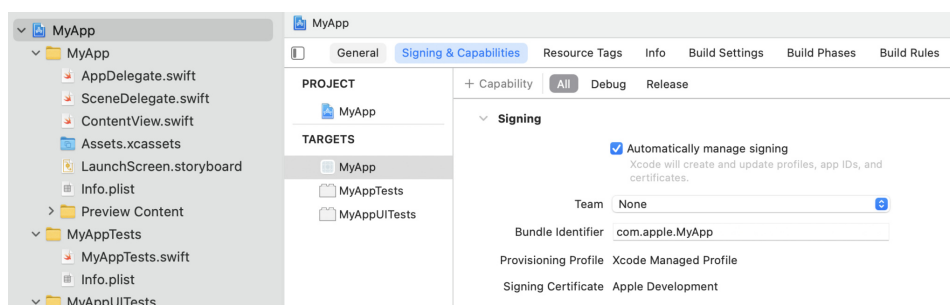


Remove unnecessary exceptions On websites where you no longer receive ATS runtime errors, you can remove those exceptions. Locate “App Transport Security Settings” in your Info.plist and click the “-” icon to remove the exceptions in question.

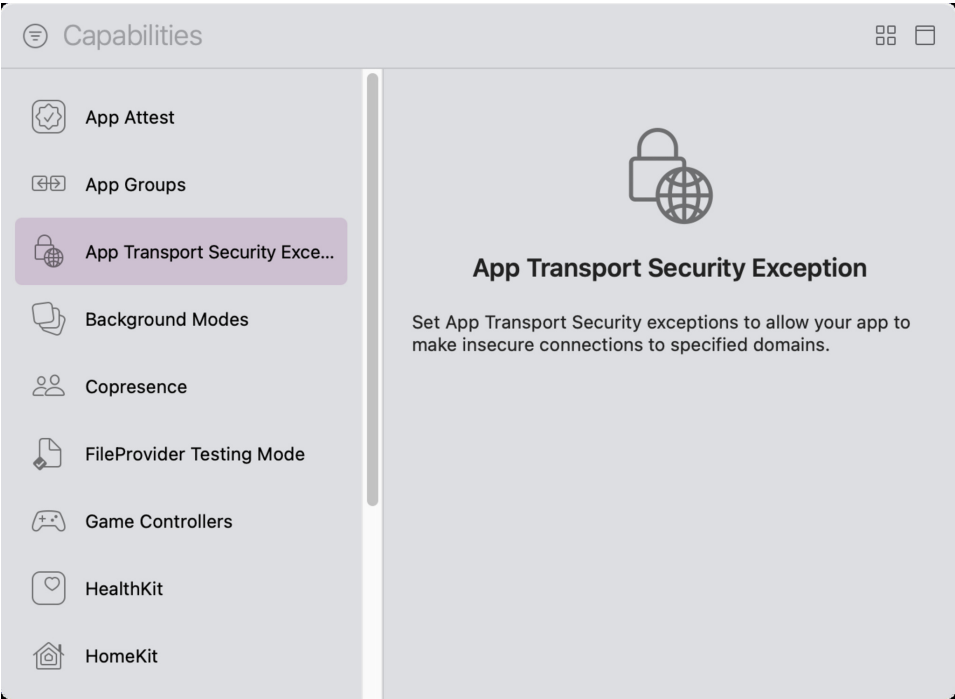
Configure exception domains

If your app still needs to make insecure connections to specific domains, you can configure ATS exceptions for just those domains.

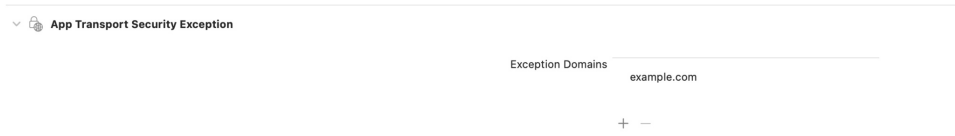
- Add Exception Domains directly to your app’s Info.plist or in the project editor. Navigate to “Signing & Capabilities” and choose the “+ Capability” option.



- Select “App Transport Security Exception” from the list.



- This will add an “App Transport Security Exception” section to your capabilities:



- Click the “+” icon to add domains that your app needs to connect to insecurely.
- Enter a domain here to connect over HTTP to that domain and its subdomains. If you need to alter these settings, you can make changes directly in your Info.plist.

▼ App Transport Security Settings	↕	Dictionary	(1 item)
▼ Exception Domains	↕	Dictionary	(1 item)
▼ example.com		Dictionary	(2 items)
NSExceptionAllowsInsecureHTTPLoads		Boolean	1
NSIncludesSubdomains		Boolean	1

Configure framework and class exceptions

On rare occasion, you still may need to make an insecure connection to an unknown domain. In this case, there are two broader exceptions you can consider offering.

- If your app needs to make insecure connections through WKWebView, add “Allows Arbitrary Loads In Web Content” to your Info.plist:

▼ App Transport Security Settings	↕	Dictionary	(1 item)
Allow Arbitrary Loads in Web Content	↕	Boolean	YES

- If your app needs to make insecure connections through AVFoundation, add “Allows Arbitrary Loads for Media”:

Discover

These exceptions will ensure that your app only makes insecure connections via AVFoundation or WKWebView, leaving the rest of your app protected by ATS. Because these are relatively broad exceptions, however, they will allow every part of your app that uses AVFoundation or WKWebView to make insecure connections which can be intercepted and inspected.

Keep your app secure

People want to trust your app, and ATS can help you build that trust by handling their data responsibly while in transit. To get the most out of ATS:

- Make sure that your app connects to servers over HTTPS instead of HTTP.
- Tailor your ATS exceptions to your app as closely as possible.
- Periodically review your exceptions to check whether servers have started supporting HTTPS or your app no longer needs to connect those servers to make insecure connections.

Resources



WWDC15
Networking with NSURLSession

- [NSAppTransportSecurity >](#)
- [Preventing Insecure Network Connections >](#)
- [NSErrorExceptionDomains >](#)
- [NSAllowsArbitraryLoadsForMedia >](#)
- [NSAllowsArbitraryLoadsInWebContent >](#)
- [WKWebView >](#)
- [Learn more about App Transport Security on the Developer Forums >](#)



Developer	News and Updates	Fine-tune your App Transport Security settings		
Discover	Design	Develop	Distribute	Support
iOS	Human Interface Guidelines	Xcode	Developer Program	Articles
iPadOS	Resources	Swift	App Store	Developer Forums
macOS	Videos	Swift Playgrounds	App Review	Feedback & Bug Reporting
tvOS	Apple Design Awards	TestFlight	Mac Software	System Status
watchOS	Fonts	Documentation	Apps for Business	Contact Us
Safari and Web	Accessibility	Videos	Safari Extensions	Account
Games	Localization	Downloads	Marketing Resources	
Business	Accessories		Trademark Licensing	
Education				
WWDC				App Store Connect

