© Developer Q Distribute News Discover Design Develop Support Account **Developer Forums** Q Search by keywords or tags

Using the Multicast Networking Additional Capability

This thread has been locked by a moderator.

© 16k

for doing this and I'm sharing them here for the benefit of all.

Greetings All I've received a number of requests for help from folks who've been granted access to the multicast entitlement

There are actually two processes involved here: • In the new process you add the Multicast Networking additional capability to your App ID and then create a provisioning profile based on

that. • In the old process you add this additional capability to your *provisioning profile* in an Additional Entitlements step.

Each process is covered in a follow-up post below.

If you have follow-up questions about this, please put them in a new thread here on DevForums and tag it with both *Network* and *Entitlements*.

Finally, since I wrote these instructions Apple has published official documentation about this process, in the form of Developer Account Help > Reference > Provisioning with managed capabilities.

Share and Enjoy Quinn "The Eskimo!" @ Developer Technical Support @ Apple

Entitlements

Network

 2020-10-08 Expanded to cover the new process. • 2020-10-06 Made minor tweaks to the No Additional Entitlements Page section.

2020-09-30 First posted.

process are compatible with manual code signing but I don't cover those details here.

New Process

each of those steps in detail.

Apple Recommended

These instructions were written for Xcode 13.2.1, running on macOS 12.1, and the developer web site at the time of posting. Mac Note This basic process also works for macOS additional capabilities. However, there are a few points where the process differs, and I've called them out with Mac-specific notes like this one. The most important of these is that the Multicast Networking additional capability is not applicable on the Mac. macOS does not implement local network privacy. These Mac-specific notes are here to help you work with other

Preparatory Work

1. Make sure you have permission to create App IDs and provisioning profiles. If not, coordinate with the folks on your team who do. See App Store Connect Help > Role permissions for more information about roles and permissions. 2. Go to the Account page on the developer web site. 3. Pull down the menu at the top right and confirm that you're logged in to the right team.

Adopt an Explicit App ID To determine whether your app already has an explicit App ID:

2. The Provision Profile field should say Xcode Managed Profile. Click the little info button next to that. In the popup you'll see an App ID field

- whose value contains a * if you're using a wildcard App ID or your bundle ID if you're using an explicit App ID. Mac Note For a macOS target the Provision Profile field may say None Required. In that case you definitely don't have an explicit App ID.
- Mac Note Adding the App Groups capability won't trigger a switch to an explicit App ID for a macOS target (that's because App Groups work differently on the Mac). To force this switch for a macOS target, add the Custom Network Protocol capability. Don't forget to remove this at the end.

1. On the developer web site, edit your App ID's capabilities using the process described in Developer Account Help > Enable app capabilities. 2. On the Edit Your App ID Configuration page, switch to the Additional Capabilities tab. 3. Enable the Multicast Networking additional capability. If you don't see that capability, see No Multicast Networking Additional Capability

The next step is to enable the entitlement in your project:

Enable the Entitlement

4. Click Save and confirm that save.

1. In your app's <code>.entitlements</code> file, add a <code>com.apple.developer.networking.multicast</code> entry with a Boolean value of true.

more information about the structure of a bundle, see Placing Content in a Bundle.

Additional capabilities issued via the old process are not compatible with automatic code signing.

Store Connect Help > Role permissions for more information about roles and permissions.

3. Pull down the menu at the top right and confirm that you're logged in to the right team.

Note These instructions were written for Xcode 12.0, iOS 14.0, and the developer web site at the time of publication.

- 2. Switch back to Signing & Capabilities and build your app for the device (not the simulator!). Xcode will then notice that you're using an entitlement that's not in the current provisioning profile's allowlist but is in your App ID's allowlist. It responds to this by rebuilding the provisioning profile. Yay!
- 3. In Terminal, run the first command below to confirm that your app was signed with the entitlement. 4. Still in Terminal, run the second command below to confirm that your profile allows use of the entitlement. % codesign -d --entitlements - /path/to/your.app | grep -A 2 "com.apple.developer.networking.multicast"
- <key>com.apple.developer.networking.multicast</key> <true/>

No Multicast Networking Additional Capability If, in step 3 of Configure Your App ID, you don't see the Multicast Networking additional capability, check the following: • That you're not set up to use the old process, as described in the Old Process post below. If you are, follow those instructions instead. • That you're logged in to the right team (step 3 of *Preparatory Work*). This is the most common cause of this problem. If this doesn't help, wait a day and try again. It can take time for this access to propagate through our systems. If you still can't get it working, run through the process again taking a screen shot at each step. Then get back in touch with us by replying to the email that notified you of your request's approval, making sure to include your follow-up number in the response.

To use the multicast entitlement (com.apple.developer.networking.multicast) you must manually create a provisioning profile that has this additional capability, then configure your app's target to use that profile, then actually enable the entitlement. The following sections



Create a Provisioning Profile

2. On the left, click Profiles.

9. Click Download.

Configure Your Target

3. Click the add (+) button at the top.

5. Select your App ID and click Continue.

Before creating your profile, you must prepare the ground:

2. Go to the Account page on the developer web site.

described each of those steps in detail.

Preparatory Work

To create a provisioning profile that includes the multicast additional capability: 1. In the Account page on the developer web site, navigate to the Certificates, Identifiers & Profiles section.

4. If your app does not already have an explicit App ID, create one for it. See Developer Account Help > Register an App ID for instructions.

6. Select the certificates and devices as you would for any other profile. 7. You'll now find yourself on the Additional Entitlements page. Select Multicast Networking from the popup and click Continue. If you don't see the Additional Entitlements page, or you don't see Multicast Networking listed there, see No Additional Entitlements Page below. 8. Give the profile a name and click Continue.

3. In the project editor, select your target on the left.

6. Select your profile from the Provisioning Profile popup.

4. Switch to the Signing & Capabilities tab.

5. Disable "Automatically manage signing".

- To configure your app's target to use the profile: 1. Locate the dowloaded profile in the Finder. 2. Drop it on to Xcode.
- **Enable the Entitlement** To enable the entitlement in your app:
- % codesign -d --entitlements :- /path/to/your.app | grep -A 1 "com.apple.developer.networking.multicast" <key>com.apple.developer.networking.multicast</key> <true/>

No Additional Entitlements Page

<key>com.apple.developer.networking.multicast</key> <true/>

• That you're not set up to use the new process, as described in the New Process post above. If you are, follow those instructions instead.

To use the multicast entitlement (com.apple.developer.networking.multicast) first ensure that you have an explicit App ID, then enable

the Multicast Networking additional capability on that App ID, and finally enable the entitlement in your project. The following sections described

These instructions assume you're building a typical app using Xcode with automatic code signing. Additional capabilities issued via the new

Mac Note This basic process also works for macOS additional capabilities. However, there are a few points where the process differs, and I've

If this doesn't help, wait a day and try again. It can take time for this access to propagate through our systems. If you still can't get it working, run through the process again taking a screen shot at each step. Then get back in touch with us by replying to the email that notified you of your request's approval, making sure to include your follow-up number in the response. Posted 2 years ago by (*) eskimo (†)

• That you're logged in to the right team (step 3 of *Preparatory Work*). This is the most common cause of this problem.

If, in step 7 of *Create a Provisioning Profile*, you don't see the Additional Entitlements page, check the following:

Before getting started, you must prepare the ground: 1. Make sure you have permission to create App IDs and provisioning profiles. If not, coordinate with the folks on your team who do. See App Store Connect Help > Role permissions for more information about roles and permissions. 2. Go to the Account page on the developer web site.

whose value contains a * if you're using a wildcard App ID or your bundle ID if you're using an explicit App ID.

These instructions were written for Xcode 13.2.1, running on macOS 12.1, and the developer web site at the time of posting.

process are compatible with manual code signing but I don't cover those details here.

- **Configure Your App ID** Once you have an explicit App ID, it's time to enable the Multicast Networking additional capability on it. To do this: 1. On the developer web site, edit your App ID's capabilities using the process described in Developer Account Help > Enable app capabilities.
- 3. In Terminal, run the first command below to confirm that your app was signed with the entitlement. 4. Still in Terminal, run the second command below to confirm that your profile allows use of the entitlement. % codesign -d --entitlements - /path/to/your.app | grep -A 2 "com.apple.developer.networking.multicast"

If, in step 3 of Configure Your App ID, you don't see the Multicast Networking additional capability, check the following:

Mac Note For a Mac app use /path/to/your.app/Contents/embedded.provisionprofile as the path in the second command. For more information about the structure of a bundle, see Placing Content in a Bundle. No Multicast Networking Additional Capability

• That you're not set up to use the old process, as described in the Old Process post below. If you are, follow those instructions instead.

Add a Comment

This site contains user submitted content, comments and opinions and is for informational purposes only. Apple disclaims any and all liability for the acts, omissions and conduct of any third parties in connection with or related to your use of the site. All postings and use of the content on this site are subject to the Apple Developer Forums Participation

Forums Developer **Topics & Technologies** Resources

Augmented Reality Video Partner Program Support Business Security Bounty Program Support Articles Design Security Research Device Program Contact Us Distribution **Bug Reporting Events** Education System Status App Accelerators Fonts App Store Awards Games Account Apple Design Awards Health & Fitness Apple Developer Apple Developer Academies In-App Purchase App Store Connect Entrepreneur Camp Localization

Safari & Web News and Updates To view the latest developer news, visit

Terms of Use

Note Almost everyone should be using the new process, but I've left the old process post in place just in case there's still a few old process folks around. It also makes a good reference for folks who are using different additional capabilities, ones that require the old process.

IMPORTANT Xcode 15 beta supports additional capabilities in the Signing & Capabilities editor. For the details, see the discussion of 27253063 in the Xcode 15 Beta Release Notes. Hopefully this will make this document largely obsolete (-:

let myEmail = "eskimo" + "1" + "@" + "apple.com" Version History: • 2022-06-12 Added a callout to the change in Xcode 15 beta. Fixed a broken link. Made other minor editorial changes. • 2022-01-27 Updated the New Process section with some Mac-specific notes. Added a link to the official documentation for this process. • 2021-05-11 Renamed everything to better match the terms used by the developer web site for the new process. • 2021-03-17 Clarified why I've left the Old Process post in place. Rewrote the New Process post to use automatic code signing.

Reply Add a Comment

Before getting started, you must prepare the ground:

1. In Xcode, go to the Signing & Capabilities editor.

Configure Your App ID Once you have an explicit App ID, it's time to enable the Multicast Networking additional capability on it. To do this:

below.

App ID will 'stick'.

Mac Note Xcode won't do this profile rebuild for a macOS target so, to force a profile rebuild, add some other capability. I've found that Personal VPN works well for this. Don't forget to remove this at the end.

[Key] com.apple.developer.networking.multicast [Value] [Bool] true

Add a Comment Replies **Old Process**

1. In your app's <code>.entitlements</code> file, add a <code>com.apple.developer.networking.multicast</code> entry with a Boolean value of true. 2. Build your app for the device (not the simulator!). 3. In Terminal, run the following command to confirm that your app was signed with the entitlement:

• That you chose to create an iOS App Development profile (step 4 of *Create a Provisioning Profile*).

Add a Comment

New Process

each of those steps in detail.

Adopt an Explicit App ID

To determine whether your app already has an explicit App ID:

1. In Xcode, go to the Signing & Capabilities editor.

called them out with Mac-specific notes like this one. The most important of these is that the Multicast Networking additional capability is not applicable on the Mac. macOS does not implement local network privacy. These Mac-specific notes are here to help you work with other additional capabilities, ones that are supported on the Mac. You will need to adjust the instructions accordingly, for example, replace com.apple.developer.networking.multicast with the entitlement associated with the additional capability you're using. **Preparatory Work**

2. The Provision Profile field should say Xcode Managed Profile. Click the little info button next to that. In the popup you'll see an App ID field

2. On the Edit Your App ID Configuration page, switch to the Additional Capabilities tab. 3. Enable the Multicast Networking additional capability. If you don't see that capability, see No Multicast Networking Additional Capability

1. In your app's <code>.entitlements</code> file, add a <code>com.apple.developer.networking.multicast</code> entry with a Boolean value of true. 2. Switch back to Signing & Capabilities and build your app for the device (not the simulator!). Xcode will then notice that you're using an entitlement that's not in the current provisioning profile's allowlist but is in your App ID's allowlist. It responds to this by rebuilding the provisioning profile. Yay! Mac Note Xcode won't do this profile rebuild for a macOS target so, to force a profile rebuild, add some other capability. I've found that Personal VPN works well for this. Don't forget to remove this at the end.

below.

4. Click Save and confirm that save.

The next step is to enable the entitlement in your project:

Enable the Entitlement

% security cms -D -i /path/to/your.app/embedded.mobileprovision | grep -A 1 "com.apple.developer.networking.multicast" <key>com.apple.developer.networking.multicast</key> <true/>

• That you're logged in to the right team (step 3 of *Preparatory Work*). This is the most common cause of this problem. If this doesn't help, wait a day and try again. It can take time for this access to propagate through our systems. If you still can't get it working, run through the process again taking a screen shot at each step. Then get back in touch with us by replying to the email that notified you of your request's approval, making sure to include your follow-up number in the response.

Agreement. **Platforms** iOS Accessibility iPadOS Accessories **App Extensions** macOS tvOS App Store Audio & Video watchOS Tools Swift SwiftUI SF Symbols Swift Playgrounds TestFlight

Copyright © 2023 Apple Inc. All rights reserved.

Certificates, IDs, & Profiles Feedback Assistant

License Agreements

Posted 2 years ago by (*) eskimo (†)

(com.apple.developer.networking.multicast) but are having problems actually enabling it in their project. I wrote up some instructions

Posted 2 years ago by (2) eskimo (1)

To use the multicast entitlement (com.apple.developer.networking.multicast) first ensure that you have an explicit App ID, then enable the Multicast Networking additional capability on that App ID, and finally enable the entitlement in your project. The following sections described These instructions assume you're building a typical app using Xcode with automatic code signing. Additional capabilities issued via the new additional capabilities, ones that are supported on the Mac. You will need to adjust the instructions accordingly, for example, replace com.apple.developer.networking.multicast with the entitlement associated with the additional capability you're using.

If you're using a wildcard App ID you must switch to using an explicit one. The easiest way to do this is to add the App Groups capability. Xcode responds to this by creating an explicit App ID for your app. If you have no use for App Groups just remove the capability afterwards; the explicit

% security cms -D -i /path/to/your.app/embedded.mobileprovision | grep -A 1 "com.apple.developer.networking.multicast" Mac Note For a Mac app use /path/to/your.app/Contents/embedded.provisionprofile as the path in the second command. For

IMPORTANT These instructions assume you're building a typical app using Xcode. This will switch your app to use manual code signing.

Posted 2 years ago by (2 eskimo)

1. Make sure you have permission to create provisioning profiles. If not, you'll need to coordinate with the folks on your team who do. See App

4. Select iOS App Development and click Continue (you can repeat this process for a distribution profile later on).

4. Still in Terminal, run the following command to confirm that your profile allows use of the entitlement: % security cms -D -i /path/to/your.app/embedded.mobileprovision | grep -A 1 "com.apple.developer.networking.multicast"

Mac Note For a macOS target the Provision Profile field may say None Required. In that case you definitely don't have an explicit App ID. If you're using a wildcard App ID you must switch to using an explicit one. The easiest way to do this is to add the App Groups capability. Xcode responds to this by creating an explicit App ID for your app. If you have no use for App Groups just remove the capability afterwards; the explicit App ID will 'stick'. Mac Note Adding the App Groups capability won't trigger a switch to an explicit App ID for a macOS target (that's because App Groups work differently on the Mac). To force this switch for a macOS target, add the Custom Network Protocol capability. Don't forget to remove this at the end.

3. Pull down the menu at the top right and confirm that you're logged in to the right team.

[Key] com.apple.developer.networking.multicast [Value] [Bool] true

Programs Documentation Apple Developer Program Curriculum Apple Developer Enterprise Program Downloads App Store Small Business Program Forums MFi Program Videos News Partner Program Tech Talks Maps & Location WWDC Machine Learning

Xcode **Xcode Cloud** Security

Privacy Policy