

[DeviceDiscoveryUI](#) / DevicePicker

Structure

DevicePicker

A SwiftUI view that displays other devices on the network, and creates an encrypted connection to a copy of your app running on that device.

📱 DeviceDiscoveryUI | 📱 SwiftUI | iOS 26.0+ | iPadOS 26.0+ | Mac Catalyst 26.0+ | tvOS 16.0+

```
@MainActor @preconcurrency
struct DevicePicker<Label, Fallback> where Label : View, Fallback : View
```

Mentioned in

📄 Connecting a tvOS app to other devices over the local network

Overview

Always display the picker as a full-screen, modal view. If the user selects a device, the system calls the closure you passed as the `onSelect` parameter. If the user cancels the picker, it silently closes.

```
DevicePicker(
    .applicationService(name: "MyAppService")) { endpoint in
    myDeviceManager.connectTo(endpoint: endpoint)
} label: {
    Text("Connect to a local device.")
} fallback: {
    Text("Not supported.")
} parameters: {
    // This example uses the default application services parameters;
    // however, you can add a NWProtocolFramer to provide application-level
```

```
// messaging.
    .applicationService
}
```

If the current device doesn't support device discovery, the system displays the fallback view instead of the device picker. Use the `DevicePickerSupportedAction` environment value to check whether the current device supports device discovery.

```
struct SettingsView: View {

    @Environment{\.devicePickerSupports} var myDevicePickerSupports
    @Binding var showDevicePicker: Bool

    var body: some View {
        if myDevicePickerSupports(.applicationService("MyAppService"),
                                   parameters: { .applicationService }) {
            Button("Select A Device") {
                // Display a device picker.
                showDevicePicker = true
            }
        }
    }
}
```

Topics

Creating a device picker

```
init(NWBrowser.Descriptor, onSelect: (NWEndpoint) -> Void, label: () ->
Label, fallback: () -> Fallback, parameters: (() -> NWParameters?))
```

Creates a view that displays the other, available devices on your local network.

Initializers

```
init<Provider>(Provider, access: DDDevicePairingAccess, onSelect: (
Provider.Endpoint) -> Void, label: () -> Label, fallback: () ->
Fallback, parameters: (() -> NWParameters?))
```

Relationships

Conforms To

Sendable, SendableMetatype, View

See Also

Selecting nearby devices



Connecting a tvOS app to other devices over the local network

Display a view in your tvOS app that lists available iOS, iPadOS, and watchOS devices that the user can connect to over their local network.

`class DDDevicePickerController`

A UIKit view that displays other devices on the network, and creates an encrypted connection to a copy of your app running on that device.

`struct DevicePickerSupportedAction`

An environment value that indicates whether the current device supports device discovery.

`NSApplicationServices`

A list of service providers and the devices that they support.