⋀ Expo 📖 Docs

Blog   ☆ Star Us on GitHub   ◐ ⌄

🔍 Search

🏠 Home

📖 **Guides**

📚 Reference

🎓 Learn

Drawer

Nesting navigators

Modals

Platform-specific Modules

Shared routes

Settings

Apple Handoff

⌄ Reference

Hooks API

Search Parameters

Redirects

Static Rendering

Async routes

Sitemap

Typed routes

Authentication

Screen tracking for analytics

Top-level src directory

Testing

Troubleshooting

Authentication in Expo Router

How to protect routes with Expo Router.

It's common to restrict certain routes to users who are not authenticated. This can be achieved in a very organized way by using React Context and Route Groups.

Consider the following project:

```
📁 app
  🗔 _layout.js
  📄 index.js
  📁 (auth)
    📄 sign-in.js
```

First, we'll setup a React Context provider that we can use to protect routes. This provider will use a mock implementation, you can replace it with your own authentication provider.

📄 context/auth.js	📋 Copy

```
import { router, useSegments } from 'expo-
import React from 'react';

const AuthContext = React.createContext(nu

// This hook can be used to access the use
export function useAuth() {
  return React.useContext(AuthContext);
```