I have multiple hooks that check for redirections such as (/src/app/auth/hooks), I want to remove these hooks and do that in the middleware.ts itself, but I want to do chaining and separate the code.  
Something like:

// Main middleware function

**export** **async** **function** middleware**(**req**:** NextRequest**)** **{**

**return** chainMiddlewares**(**req**,** **[**

authMiddleware**,**

nonAuthMiddleware

**]);**

**}**

I want to separate the middleware into two:

* one for the authPages like [‘/auth/login’, ….]
* And another middleware for the other nonAuthPages that aren’t in authPages

The current middleware.ts state is for the nonAuthMiddleware for the nonAuthPages,

I want to add another middleware for the authPages like the login, 2fa-setup, 2fa-verify to handle the redirections from there instead of doing it from the hooks.

1. Logic For the Login (pathname === ‘/auth/login’):

* If there’s no auth-token in the cookies, leave the login page, otherwise, I want to check its data by using check2FAStatusAction() that checks the ‘auth-token’ if valid, we’ll get the columns flags

**try** **{**

// Check if user is already authenticated

**const** result **=** **await** check2FAStatusAction**();**

**if** **(**result**.**success**)** **{**

**const** **{** is2FAEnabled**,** hasSetup2FA**,** is2FAVerified **}** **=** result**;**

// If 2FA is disabled, go directly to homepage

**if** **(**is2FAEnabled **===** **false)** **{**

console**.**log**(**"🔍 2FA is disabled, redirecting to homepage"**);**

router**.**push**(**"/"**);**

**return;**

**}**

// If 2FA is set up and verified, go to homepage

**if** **(**hasSetup2FA **===** **true** **&&** is2FAVerified **===** **true)** **{**

console**.**log**(**

"🔍 2FA is set up and verified, redirecting to homepage"

**);**

router**.**push**(**"/"**);**

**return;**

**}**

// If 2FA is set up but not verified, go to setup

**if** **(**hasSetup2FA **===** **true** **&&** is2FAVerified **===** **false)** **{**

console**.**log**(**

"🔍 2FA is set up but not verified, redirecting to 2FA setup"

**);**

router**.**push**(**"/auth/2fa-setup"**);**

**return;**

**}**

// Otherwise, go to 2FA verify (user needs to set up 2FA)

console**.**log**(**"🔍 Redirecting to 2FA verify"**);**

router**.**push**(**"/auth/2fa-verify"**);**

**}** **else** **{**

console**.**log**(**

"🔍 No valid authentication found, staying on login page"

**);**

// No valid authentication, stay on login page

**}**

**}** **catch** **(**error**)** **{**

console**.**error**(**"🔍 Error checking authentication status:"**,** error**);**

// On error, stay on login page

**}**

1. Logic For the 2fa-setup (pathname === ‘/auth/2fa-setup):

* If there’s no auth-token in the cookies, redirect to login, otherwise, I want to check its data by using check2FAStatusAction() that checks the ‘auth-token’ if valid, we’ll get the columns flags

**try** **{**

// Check if user is already authenticated

**const** result **=** **await** check2FAStatusAction**();**

**if** **(**result**.**success**)** **{**

**const** **{** is2FAEnabled**,** hasSetup2FA**,** is2FAVerified **}** **=** result**;**

// If 2FA is disabled, go directly to homepage

**if** **(**is2FAEnabled **===** **false)** **{**

console**.**log**(**"🔍 2FA is disabled, redirecting to homepage"**);**

router**.**push**(**"/"**);**

**return;**

**}**

// If 2FA is set up and verified, go to homepage

**if** **(**hasSetup2FA **===** **true** **&&** is2FAVerified **===** **true)** **{**

console**.**log**(**

"🔍 2FA is set up and verified, redirecting to homepage"

**);**

router**.**push**(**"/"**);**

**return;**

**}**

// If 2FA is set up but not verified, go to setup

**if** **(**hasSetup2FA **===** **true** **&&** is2FAVerified **===** **false)** **{**

**if (**secret2FA**) {**

router**.**push**(**"/auth/2fa-verify"**);**

**return;**

**} else {**

console**.**log**(**"🔍 staying on the current 2fa-setup"**);**

**}**

**}**

**}**

**}** **catch** **(**error**)** **{**

console**.**error**(**"🔍 Error checking authentication status:"**,** error**);**

// On error, stay on login page

**}**

1. Logic For the 2fa-verify (pathname === ‘/auth/2fa-verify):

* If there’s no auth-token in the cookies, redirect to login, otherwise, I don’t want to check the status here, I don’t want to make a headache to the server