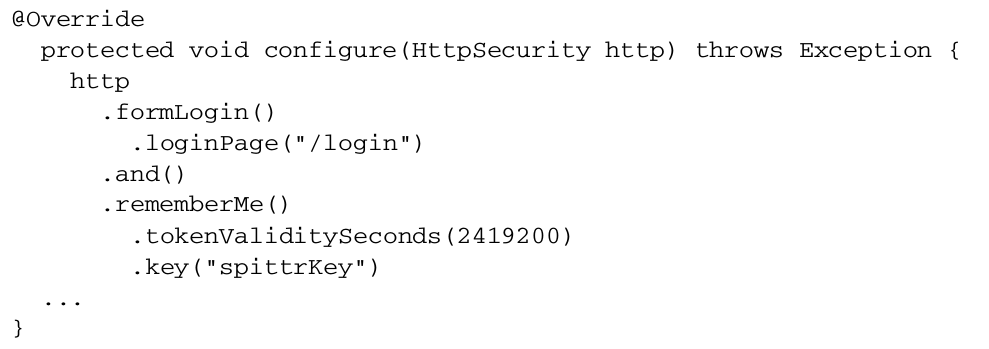
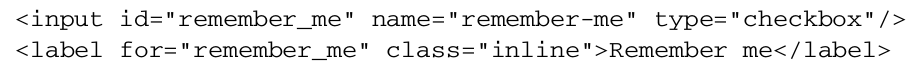
***Enabling remember-me functionality***

It’s important for an application to be able to authenticate users. But from the user’s perspective, it’d be nice if the application didn’t always prompt them with a login every time they use it. That’s why many websites offer remember-me functionality, so that you can log in once and then be remembered by the application when you come back to it later. Spring Security makes it easy to add remember-me functionality to an application. To turn on remember-me support, all you need to do is call rememberMe() on the HttpSecurity passed into configure() :

Here, in addition to turning on remember-me functionality, a bit of special configuration has also been added. By default, a remember-me token is stored in a cookie that’s valid for up to two weeks. But this example specifies that the token should stay valid for up to four weeks (2,419,200 seconds).

The token that’s stored in the cookie is made up of the username, password, an expiration date, and a private key—all encoded in an MD5 hash before being written to the cookie. By default, the private key is SpringSecured , but this example sets it to spitterKey to make it specific to the Spittr application.

Simple enough. Now that the remember-me functionality is enabled, you’ll need a way for users to indicate that they’d like the application to remember them. For that, the login request will need to include a remember-me parameter. A simple check box in the login form ought to do the job:



Just as important as being able to log in to an application is the ability to log out. This is especially true if you’ve enabled remember-me; otherwise the user would be logged into the application forever.