From one simple feature you can see how I've had so much to test for already. Apply this throughout the web application and you can start to see why it takes time, patience and hard work.

- **After creating an application, how does the login flow actually work? And when I "disconnect" the application from my profile. Is the token invalidated?** Are there new return_uri parameters used and how do they work? One little "trick" is you may discover some companies whitelist certain domains for debugging/testing. Try theirdomain.com as the redirectUri as well popular CDNs such as amazonaws.com, .aws.amazon.com. http://localhost/ is common also but wouldn't affect all users (*they'd have to be running something on their machine*)

- **Does the wiki/help docs reveal any information on how the API works?** (*I once ran into a problem where I could leak a users token but I had no idea how to use it. The wiki provided information on how the token was authenticated and I was able to create P1 impact*). API docs also reveal more API endpoints, plus keywords for your wordlist you're building for this target.

- **Can I upload any files such as an application image?** Is the filtering the same as updating my account information or is it using a **different codebase**? Just because uploading your profile photo on example.com wasn't vulnerable doesn't mean different code is used when uploading a profile photo on developer.example.com

- **Can I create a separate account on the developer site or does it share the same session from the main domain?** What's the login process like if so? Sometimes you can login to the developer site (developer.example.com) via your main session (www.example.com) there will be some type of token exchange handled by a redirect. Enter that open url redirect you've probably discovered by now. If it's a brand new account then re-enter the process of seeing what's reflected & where etc. **I actually prefer** when you need to sign up for a new account because it means there's more than likely going to be different code being used, resulting in a