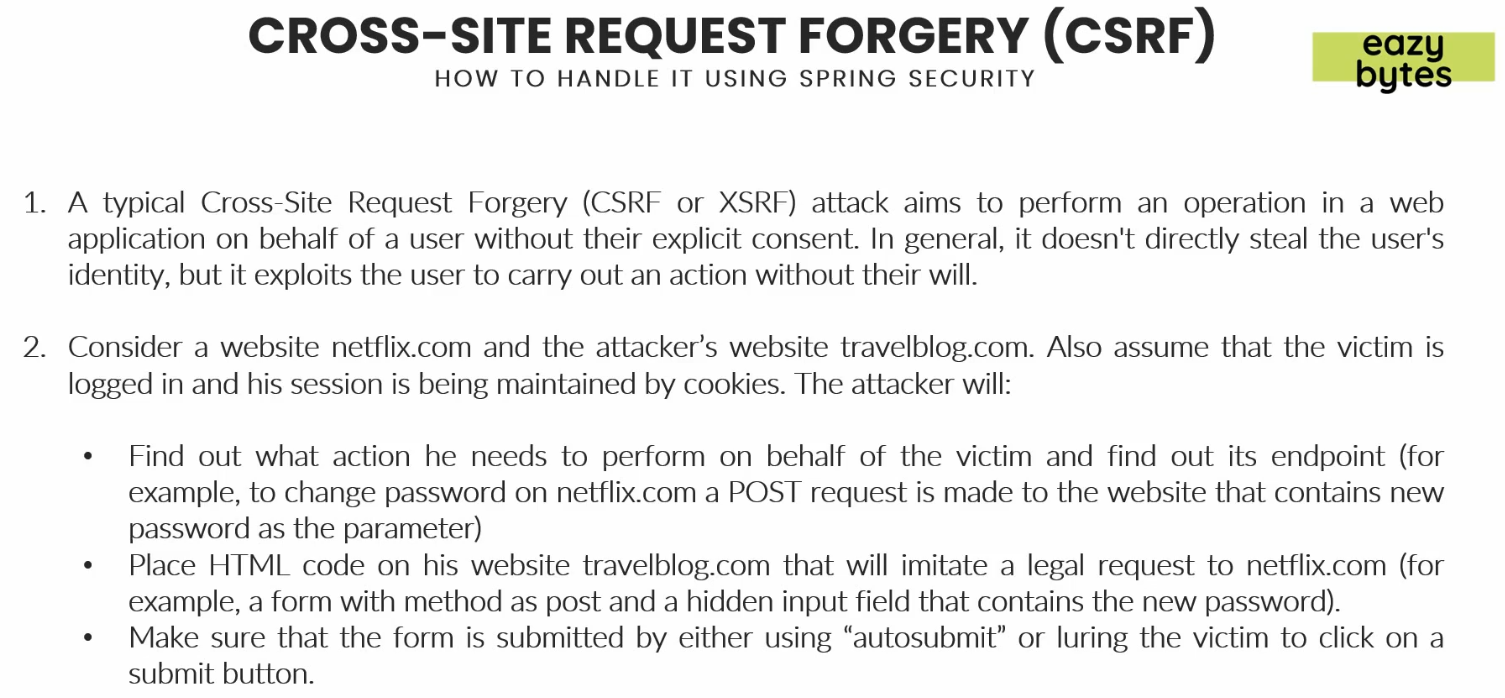
1. NOTE: first keep in mind that CSRF has nothing to do with CORS.  
   Even if your front-end is deployed on the same origin, when making destructive HTTP requests (POST, DELETE, PUT etc), the backend server will throw 403 forbidden error.
2. The reason is a typical CRSF aims to perform an operation in a web app on behalf of a logged in user without the explicit consent.
3. There are many ways a Hacker can attach other users.
4. Maybe you have taken enough measures not to steal their credentials, but CSRF is one of the smartest approach where the attackers can exploit if you don’t handle it properly.
5. 
6. There are many ways to avoid these sorts of attacks.