**Practical exercises and questions**:

In the following section there will be six exercises about Broken Access Control and Cross Site Request Forgery. We suggest you to follow the exercises in the order in which they are presented.

**Exercise 1**: In the following exercise you will experience what happens if access control can be circumvented by changing request methods.

You can find the lab here: https://portswigger.net/web-security/access-control/lab-method-based-access-control-can-be-circumvented

**Objective**: The goal of the exercise is log into the application and promote yourself to administrator. Keep in mind that this lab implements access controls based partly on the HTTP method of requests. You can familiarize with the admin panel by logging in with: Username: administrator – Password: admin. Use these credentials to access the application and perform the attack: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how a legitimate request works. Try to mess with different parameters and methods. The access control mechanism might be vulnerable.*

**Exercise 2**: In the following exercise you will experience what happens if access control is not implemented the right way.

You can find the lab here: https://portswigger.net/web-security/access-control/lab-multi-step-process-with-no-access-control-on-one-step

**Objective**: The goal of the exercise is log into the application and promote yourself to administrator. Keep in mind that this lab has an admin panel with a flawed multi step process for changing a user's role. You can familiarize with the admin panel by logging in with: Username: administrator – Password: admin. Use these credentials to access the application and perform the attack: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how the different steps of a legitimate request work. You might be able to forge your own request and bypass the access control mechanism.*

**Exercise 3**: In the following exercise you will experience what happens if access control can be circumvented by directly pointing at a URL.
You can find the lab here: https://portswigger.net/web-security/access-control/lab-referer-based-access-control

**Objective**: The goal of the exercise is log into the application and promote yourself to administrator. Keep in mind that this lab controls access to certain admin functionalities based on the Referer header. You can familiarize with the admin panel by logging in with: Username: administrator – Password: admin. Use these credentials to access the application and perform the attack: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how a legitimate request work. You might be able to bypass the access control by directly pointing at the right URL.*

**Exercise 4**: In the following exercise you will experience what happens if tokens are duplicated in cookies and can be easily injected.
You can find the lab here: https://portswigger.net/web-security/csrf/bypassing-token-validation/lab-token-duplicated-in-cookie

**Objective**: The goal of the exercise is to change the viewer's email address using some HTML that performs a CSRF attack. The only thing you know is that the email change functionality is vulnerable to CSRF. It attempts to use the insecure "double submit" CSRF prevention tecnique. Use these credentials to access the application: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how a legitimate request work. Which are the key parameters for the request? Can you manipulate some of those in* *some way? Explore the different functionalities of the application, you may find something useful. Use the exploit server to deliver the attack.*

**Exercise 5**: In the following exercise you will experience what happens if validation is not implemented in the correct way.

You can find the lab here: https://portswigger.net/web-security/csrf/bypassing-referer-based-defenses/lab-referer-validation-depends-on-header-being-present

**Objective**: The goal of the exercise is to change the viewer's email address using some HTML that performs a CSRF attack. The only thing you know is that the email change functionality is vulnerable to CSRF. It attempts to block cross domain requests but has an insecure fallback. Use these credentials to access the application: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how a legitimate request work. Is everything necessary for the request? Try to mess with the different parts of the request, you may find something interesting. Use the exploit server to deliver the attack.*

**Exercise 6**: In the following exercise you will find another example of what happens if validation is not implemented in the correct way.

You can find the lab here: https://portswigger.net/web-security/csrf/bypassing-referer-based-defenses/lab-referer-validation-broken

**Objective**: The goal of the exercise is to change the viewer's email address using some HTML that performs a CSRF attack. The only thing you know is that the email change functionality is vulnerable to CSRF. It attempts to detect and block cross domain requests but the detection mechanism can be bypassed. Use these credentials to access the application: Username: wiener – Password: peter

**A suggestion in case you are lost**: *Try to understand how a legitimate request work. Is the validation happening in the right way? Try to mess with the string in the Referer Header, you may find something interesting. Use the exploit server to deliver the attack.*

**Questions**:

In this little section you will find some simple questions to test your comprehension of the topic.

1. Describe how broken access control problems can be mitigated

2. Describe how a CSRF attack can be prevented