WAS 2015: TP 3

1. (File: guestbook.php, guestbookleavemessage.php. You must create file message.txt.) Use the application: to which kind of vulnerability suffers the application? Demonstrate with an attack that disables the input elment of the guestbook. Mention 2 ways in which this attack could be prevented and implement them in 2 versions of guestbook: guestbookA.php and guestbookB.php.

2. (File: guestbook2.php, guestbookleavemessage2.php). This application uses a standard php function for sanitization, however the following attacker code can be executed: **this is a nice message");alert("this is attacker code");console.log("** Try the attack and explain why it works in spite of sanitization. How do you correct this vulnerability?

3. (File: xsrf.php, simple.php) Explain which code should attackerGadget.js have to produce a CSRF attack and answer:

   - which is the CSRF attack?
   - can the attack take place if the gadget is in an iframe? Justify your answer.
   - can the attack take place if the cookie is httponly? Justify your answer.

   Using tokens, prevent CSRF attacks in this application.

4. (File: xsrf.php, simple.php, it depends on previous exercise.) Having implemented a defense against CSRF attacks, explain how attackerGadget.js could mount an XSS attack to circunvent the CSRF defense and produce an CSRF attack. After implemented the attack, explain how do you prevent this.

5. (No file, you need a browser that supports CSP) Audit your browser https://browseraudit.com/ to verify that it supports CSP. Write an application that uses CSP to mitigate XSS attacks. Read the following article http://www.cse.chalmers.se/ andrei/dimva15.pdf and find a browser extension that works with your application and another that does not work. Explain the reasons.