Exercise11: PHP Security

Objectives:

Get familiar with PHP security concepts

Use a static analysis tool called RIPS to find vulnerabilities in PHP file

Work with your group (or by yourself). Each group is to upload only one submission.

1 Warm Up: Try Some Examples

First scan this section. You will need to create a report and also a folder with fixed examples.

1. First, open blackboard, go to Course Contents, and then download exercise11.zip file into your workspace (U:\workspace or something like that!). Then, unzip.
2. Download RIPS

   RIPS is a tool written in PHP to find vulnerabilities in PHP applications using static code analysis. You can download from http://rips-scanner.sourceforge.net/. Alternatively, you may find it under exercise11.zip file under the name rips.zip. Unzip "rips.zip" file Create a directory called "rips" under the www folder of your uWamp (recall from exercise 6) installation.

   Place the unzipped contents of the "rips.zip" into the "rips" folder
3. Now if you go to http://localhost:8080/rips/ using your browser, you should see rips front page getting loaded successfully.
4. In order to run RIPS to find vulnerabilities, you set the path you want to analyze in "path / file:" and Let the "vuln type:" remains "All".
5. Copy the examples directory into www folder. There are four sub folders there. Run the RIPS tool on each of these examples separately to find their vulnerability (or vulnerabilities). Note that if you click on the "?" on the result page, it shows details of the specific vulnerability and also how to handle it.
6. Create a REPORT document. Name vulnerabilities of each of these examples and explain them in details (along with a possible fix for each of them) in the first section of your report file. Remember that these examples might not be runnable because they are code excerpts to just show you different vulnerability types. RIPS is a static analysis tool, so it does not run the code, but only manipulates the source code.
7. Next, fix all the issues for all examples and save the fixed version of each example into a new folder called "examples-fixed".
8. You will need to submit the REPORT and the examples-fixed folder.

2 ".htaccess"

Create a file named ".htaccess" in the www folder of UwaAmp. Leave the file empty. If you are on windows, create a text file, open it with notepad and then "Save as" it as ".htaccess". Now, direct your browser to http://localhost:8080. Click on examples folder. You should be able to see the contents of the examples folder. Now write "`Options -Indexes`" into .htaccess file and save it. Again, direct your browser to http://localhost:8080 and try to access the contents of examples folder. What happened? You should not be able to see the directory listing any more. It is always a good practice to set up the .htaccess file properly to prevent attacks such as brute force attack.

3 Exercise 6 vulnerabilities

Now, copy the folder called ex06 into www folder (this is the submission for exercise 6 from one of the groups that we use as a subject for security testing in this exercise). Run RIPS on ex06 and all its subfolders. How many vulnerabilities can RIPS find? What are the most common types of vulnerabilities found? Create a table in your report file with three columns. The first column shows the name of the vulnerability. The second column shows the location of it (sourcefile path+line number). The third column describes it and proposes a fix. In addition to all types of vulnerabilities that RIPS can find, inspect the application and all its source files for the following vulnerabilities:

[ ] Vulnerabilities associated with weak typing (e.g. ==, arrays etc)
[ ] Unicode Encoding Attack
[ ] Web Parameter Tampering

You may find these resources useful:

https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet
https://www.owasp.org/index.php/Web_Parameter_Tampering
https://www.owasp.org/index.php/Unicode_Encoding

FIX AT LEAST SIX VULNERABILITIES. Add these to the table as well and fix the code. Make sure the code runs properly given the fixes.

4 Submission

Your submission has to include the following items:

1. A report that details and explains all vulnerabilities found in the warm-up examples.
2. Fixed version of source codes of warm-up examples.

3. A table in the report that details and explains all vulnerabilities and fixes for the for the source files of the given exercise 06 submission. Once again, you need to consider all vulnerability types detectable by RIPS plus all vulnerabilities associated with PHP being a weakly typed language, unicode encoding attack, and web parameter tampering, if any.
4. Fixed AT LEAST SIX vulnerabilities for the source codes of the given exercise 06 submission.

5. Place everything into one .zip file. Include your group number in the file name.