CSE 5382-001: Secure Programming
Spring 2019, © TJ, UTA 2018

Programming Assignment 4 – Rev A
Due: May 3, 2018, 23:59 UTA time
Finding, Exploiting, and Fixing Vulnerabilities in Web Apps

Overview

There is lots of bad code out there. Now that you have familiarized yourselves with static analysis tools and some of the most common software vulnerabilities found in web applications, it is a good time to exercise what you've learned and gain a practical understanding of these vulnerabilities. Rather than have you (or me) deploy a bad site, and risk (you can imagine), there is a ready-made, "bad" site: "Metasploitable".

You will first need to download the VM image from http://sourceforge.net/projects/metasploitable/files/Metasploitable2/ and run it on some virtualization software (free ones are: VMware Player and VirtualBox).

Then you will find the IP address of the VM (hint: ifconfig or ip addr), enter it into your web browser and start looking for vulnerabilities in the web services. You may use any static analysis tools (those discussed in class or any others you wish) to assist your search. You need to find an actual exploit for each vulnerability and then fix the code in order to prevent the vulnerability from reoccurring.

You should find at least 10 (of the CSE/SANS Top 25) different types of vulnerabilities you discovered in the site (VM image). You should show how to fix any 6 of these.

Submission:

You will submit a report that includes a summary of at least 10 different types of vulnerabilities you discovered along with:

- Description of the attack vector exploiting the vulnerability (you need to actually exploit the vulnerability).
- How you mitigated (fixed) the vulnerability (description/code).
- How you discovered the vulnerability (tools, code analysis).
- Screenshots will be an essential part.

Hint: The Mutillidae web service on this image (a link can be found when you enter the VM IP address in your browser) has each of the OWASP Top 10 vulnerabilities. You only need to find 5 others. Other useful tools that will help you include FireBug, Burp-Suite, Wireshark, etc. Use the hints in the application as a last resort (your analysis cannot be a regurgitation of the hints). Also, be mindful of the Security setting in the application as it can make your job harder if set too high.

In addition, please avoid simply repeating the examples shown in class using Damn Vulnerable Web Application (DVWA).

Try to address as many unique vulnerabilities as possible. If you end up addressing the same type of vulnerability more than once, use different methods for finding/exploiting it if possible.

Bonus

Identify an additional 5 vulnerabilities – get 10 bonus points (no partial credit; must identify at least 5). Show how to fix 4 of those 5 – get another 20 bonus points (no partial credit; must show how to fix at least 4).