CS 4235/6035 Lab Three
Spring, 2016
Joel Odom, Course Instructor, Evan Downing, Teaching Assistant Responsible for this Lab

There are 20 points available for this lab. This lab is due Wednesday, 20 April. Please start early! You are encouraged to work in small groups of at most three students. We require only one submission per group. Remember to list all of your group members on your submission. *Include enough writing and screen captures in your submission so that we can see clearly that you completed the lab and so that we can award partial credit where possible.*

Writing Secure Software (5 Points)

Write and demonstrate a short C or C++ program with a software vulnerability. Your vulnerability could be a buffer overflow / overread, unsafe use of a system call or of temporary files, incomplete mediation of program input, or another kind of security vulnerability.

Submit your source code of the vulnerable program, an explanation of the vulnerability (including how to compile the source code and what modifications you made to your environment like disabling ASLR, DEP, etc.), and a script to exploit the vulnerability in the program.

There are many tutorials online of how to write programs with vulnerabilities and how to exploit them.
For example: https://www.eecis.udel.edu/~bmiller/cis459/2007s/readings/buff-overflow.html

Note that in the real world you have to find the vulnerability yourself. They aren't given to you on a silver platter.
Exploits are usually pretty simple to write (maybe 5 lines of python), but locating the vulnerability in the first place takes lots of practice and experience.
We're making this assignment easier on you by requiring that you write your own vulnerable program to exploit.

Also note that ASLR (https://en.wikipedia.org/wiki/Address_space_layout_randomization) is difficult to defeat if you have no experience with exploiting systems. So you are allowed to disable ASLR and other program protection techniques like DEP (https://en.wikipedia.org/wiki/Data_Execution_Prevention) to get your exploit to work properly. Just make sure you make notes of what you disabled and modified as we asked above.

What to submit:

- Source code of vulnerable program
- Explanation of vulnerability
- Modifications you made to your environment (disabling ASLR, DEP, etc.)

- Source code of script that exploits the vulnerable program with comments inside providing a detailed step-by-step explanation of how the vulnerability is exploited.

Reverse Engineering (2.5 Points)

Kali Linux comes with a handful of reverse engineering tools (such as edb-debugger and strace) readily available for Kali. Choose at least two reverse engineering tools and submit screen captures and a write-up that demonstrate and explain how a security researcher could use those tools to reverse engineer your vulnerable program. For full credit your write up should have enough detail to convince the grader that you could probably find the vulnerability from the previous section without the source code and without debugging symbols. Please be thorough.

Studying Malware (2.5 Points)

It used to be that malware would solely perform some annoying task such as making your computer screen turn black or crashing your hard drive. But then these same people figured out that they could make money doing less: by keeping the victim's machine alive and well, they could continuously steal personal information, cause other nefarious activities across the Internet, or even sell the control of the victim's machine to other interested attackers.

Malware written for fun: https://en.wikipedia.org/wiki/Happy99
Malware written for profit: https://securityintelligence.com/dyre-wolf/

Use Google to search for some malware that has been analyzed by virustotal.com **and** malwr.com. Make sure the malware is the same sample in both reports (i.e., make sure the SHA-256 hashes are the same). Search for this malware online and find some article(s) that reports on its activities.

For example, I found some malware called Backoff POS by using a few Google searches. Then I searched virustotal.com and malwr.com for that malware to find a sample of it that has been analyzed by **both**. So I found one on report on virustotal (https://www.virustotal.com/en/file/3a40b3fcb0707e9b5ae6dd9c7b4370b101c37c0b48fa56a602a39e6d7d5d0de5/analysis/) and one on malwr.com (https://malwr.com/analysis/MTdmZDRkNmY4Yjk4NDYxOTk0NmI5NmU3YzNmYTI4MmY/). Then I found an article about Backoff POS (https://www.us-cert.gov/ncas/alerts/TA14-212A) and read about it.

Please do not use Backoff POS for your answer since I have already done the work above. I was simply showing you how to search for malware reports on the Internet.

Use the analyses from virustotal.com, malwr.com, and the article(s) to answer the questions below:

1. What's the malware's common name?
2. When was this malware **first** discovered? (use the article you found online)
3. Provide links to the malware on virustotal.com, malwr.com, and the article you found.
4. At a high-level, what does the malware do? (a few sentences)
5. At a low-level, what does the malware do? (a paragraph or two)
6. What other filenames has the malware been known as?
7. How does the malware infect the victim's computer? (i.e., email attachment, exploited vulnerability, social engineering, etc.). If there are no articles describing how the victim's computer got infected in the first place, please say so and come up with an explanation of how you would infect your victim's computer if you were the author of the malware.
8. In your own words, describe why you think this piece of software was classified as malware. What makes it malicious in your opinion?

Metasploit (10 Points)

In our first lab we used a virtual image called Metasploitable. This is an intentionally vulnerable Linux virtual machine for security training, security tools testing, and penetration testing. In this problem we are going to practice some penetration testing techniques. Metasploit is one of the most popular penetration testing tools. Using Metasploit on your Kali VM, exploit the Metasploitable VM using the following vulnerabilities:

1. UnrealIRCD IRC daemon backdoor
2. □VSFTPD backdoor
3. PHP CGI Argument Injection vulnerability
4. DistCC Daemon Command Injection vulnerability

There are lots of how-tos on how to exploit metasploitable:

- http://www.hackavision.com/2012/11/hacking-metasploitable-1-introduction.html
- https://community.rapid7.com/docs/DOC-1875

Google for yourself how to exploit these vulnerabilities in Metasploitable.
They're pretty easy to look up.

Once you've successfully been able to exploit Metasploitable, you'll be able to execute commands from Kali on the Metasploitable machine. You can check your work by performing `ls` on "/home/" and finding the "msfadmin" directory. You can also execute "hostname" and see that the hostname is that of the Metasploitable machine. Please include this screencap of you executing a command on the compromised host.

Be sure to show all that you did with words and screen captures in your report. Please make your report high-quality, complete, and understandable for grading purposes.

What to submit:

- Explanations of each vulnerability (include sources)
- Exact commands used to exploit vulnerability in Metasploitable using Kali Linux
- Screenshots of your exploits.