CSC 333 Penetration Test Report

Justyce J. Countryman

Computer Science Department, SUNY Oswego

CSC 333: Privacy / Security / Cryptography & Special Topics

Dr. James Early and Dr. Bastian Tenbergen

June 30, 2023

Performing a penetration test is an ideal method of obtaining critical information about how an attacker could get into a network and potentially cause significant damage to users in and out of the network. This project gives my crucial findings for a sample penetration test when given little information about the network, which is known as a gray box test, to prepare for a career involving cybersecurity where figuring out vulnerabilities and how attackers could exploit them are of the utmost importance (Ciampa, 2015). The sample network I found contained several systems that are known to have some vulnerability that attackers could use as threat vectors. For my project, I utilized the computer with IP address 192.168.56.102 and found a vulnerability due to the availability of port 80, which is an indication that HTTP, or Hypertext Transfer Protocol, is in use, meaning that it could be possible to target the attack through the web application connected to this IP address. The findings of this penetration test allowed for the collection of several usernames, a password that gave SSH, or Secure Shell, access with one of the usernames, and the ability to obtain root privileges, which could expose files and data that are intended to be kept private from outside users.

The approach for the attack on 192.168.56.102 did require some basic knowledge of ports, HTTP, terminal commands, SSH, and Kali Linux tools, but once the acquired tools and mastery of these cybersecurity concepts are obtained, it becomes a significant concern that users outside of the network may learn how to attack this system with ease. To begin, the first step of any penetration test is to locate which computers are on the target network, which is known as pre-engagement (*Penetration Test*, 2020). The terminal command to apply this step is `netdiscover -i eth1`, which revealed several computers in the eth1 server through the Netdiscover Kali Linux tool, including the host system with IP address 192.168.56.102. Secondly, the reconnaissance step comes into play by scanning the computer to look for any open ports that may be exploited (*Penetration Test*, 2020). Using the Nmap Kali Linux tool with the command `nmap -p- 192.168.56.102` revealed that ports 22 and 80 are open, which signify that SSH and HTTP are being used respectively. By launching Firefox through my Kali Linux account in the terminal, I could then type in the IP address in the search bar to access the vulnerable web application. The next step was to apply the Netcat Kali Linux tool to listen to port 80 on the host to ensure that the port is open, which introduces the third step in the penetration test, vulnerability assessment (*Penetration Test*, 2020). The command used to determine that the port was indeed open was `nc -nv 192.168.56.102 80`. Afterwards, the Gobuster Kali Linux tool was used to start the fourth step of the penetration test, exploitation, to locate any interesting files or directories from the web application with the terminal command `gobuster dir -u http://192.168.56.102 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x .php,.html` (Trimukhe, 2021; *Penetration Test*, 2020). With this command, a critical file known as `search1.php` was found (Trimukhe, 2021). This file was crucial to the completion of the attack because the source code of the webpage view-source:http://192.168.56.102/search1.php indicated two lines of code, `ME=about.php` and `FUZZ=contact.php`, that correspond to the "About" and "Contact" pages of the web application. After changing the URL of the "About" webpage from http://192.168.56.102/search1.php?Me=about.php to http://192.168.56.102/search1.php?me=/etc/passwd, then viewing the new source code with view-source:http://192.168.56.102/search1.php?me=/etc/passwd, several usernames were found by highlighting the lines of code that had `/bin/bash`, with one of them being a root user. The usernames found in the system include hacksudo, monali, john, and search. Continuing on, the

Nikto Kali Linux tool came into use to receive specific vulnerability details about the host system with the command `nikto -h 192.168.56.102`. The output detailed a file called `/.env`, which may contain user credentials according to Nikto. Opening the webpage http://192.168.56.102/.env did in fact uncover login credentials for a mySQL database, with hiraman as the username and MyD4dSuperH3r0! as the password. Furthermore, I created a file called `projectusernames.txt` that will hold the previously discovered usernames. The Hydra Kali Linux tool then came as useful to see if there are any successful login credentials that will give access to SSH servers on the host system via port 22 with the command `hydra -L projectusernames.txt -p MyD4dSuperH3r0! ssh://192.168.56.102:22 -vv` (Trimukhe, 2021). Hydra returned the username hacksudo and the password MyD4dSuperH3r0! as valid login credentials for the SSH server. As a result, using the command `ssh hacksudo@192.168.56.102` and then entering the valid password allowed for access into the hacksudo account (Trimukhe, 2021). For the fifth step of the penetration test, post-exploitation, I attempted to obtain root privileges for the system by taking advantage of a SUID binary file called `searchinstall.c` found in the `/search/tools` directory (Trimukhe, 2021; *Penetration Test*, 2020). The commands `echo "/bin/bash" > install`, `chmod +x install`, and `export PATH=/home/hacksudo/search/tools:$PATH` were applied to create a bash script with the name ‘install,’ give it permission to be executed, and change its path so that it would be sent into the `/search/tools` directory respectively (Trimukhe, 2021; *Penetration Test*, 2020). Finally, installing the `searchinstall` file in the `/search/tools` directory then got me into root mode for the hacksudo account (Trimukhe, 2021). To ensure I had root access, I used the `id` command, then I went to the `/root` directory by going to the home directory, then going back to the parent directory, and then changing to the `/root` directory (Trimukhe, 2021). From there, I was able to open a file called `root.txt`, a file that should only be accessible to the root user known as hacksudo, but performing a privilege escalation exploitation thanks to several vulnerabilities throughout the system gave me the ability to obtain root privileges and files. For the sixth and final step of the penetration test, reporting, it would be advantageous to not just mention the means of which an attack on the system is possible, but to also state some solutions to reduce or even eliminate the risk of losing any assets like data, files, the host system, or the entire network (*Penetration Test*, 2020). To avoid anyone easily finding the credentials to any user of the system, the `search1.php` file should be removed since its presence gets attackers a step closer to finding all of the users of the SSH server for this host system. Additionally, the http://192.168.56.102/search1.php?me=/etc/passwd webpage should have its source code updated so that the database that contains the valid usernames for the SSH server are in a more secure location, like on the root account since more layers of security would have to be surpassed. Not only that, this plan should also be used for the `/.env` file to prevent attackers from obtaining the mySQL database password. Lastly, if the attacker still manages to get into the SSH server, the `searchinstall.c` SUID binary file needs to be deleted or moved into a directory only users with root privileges can access in order to patch a vulnerability that may grant privilege escalation to unwanted users, the same plan should apply to the `searchinstall` file.

With the situation in mind about what could happen if the security of this host system is not improved, it will only be a matter of time before an attacker who has malicious intentions try to infect the system and possibly the entire network, making people who work with these systems spend time and money attempting to restore stability to the system and regain lost data and files, resulting in massive delays in work productivity. With easily accessible files that contain login credentials via HTTP and ones that give an opportunity for privilege escalation through SSH, critical vulnerabilities are established that could give attackers, even ones with little cybersecurity experience, the chance to gain control of a user account or even root privileges. Although these files provide important information for approved users, they should be protected with as many layers of security as possible. Any other protective strategy that goes along with the five principles of security, including layering, limiting, diversity, obscurity, and simplicity, will be worthwhile for all users within the network, especially when they do not have to fear their login credentials being compromised and their files getting stolen or corrupted (Ciampa, 2015).

References

Ciampa, M. (2015). *CompTIA Security+ Guide to Network Security Fundamentals*. CompTIA.

Trimukhe, A. (2021, April 26). *Vulnhub Walkthrough : hacksudo_search*. akshaytrimukhe.medium.com. https://akshaytrimukhe.medium.com/hello-today-i-am-going-to-solve-another-vulnhub-m Achine-called-hacksudo-search-df2e04224a6b

YouTube. (2020). *Conduct a Penetration Test Like a Pro in 6 Phases [Tutorial]. YouTube.com*. Retrieved June 29, 2023, from https://www.youtube.com/watch?v=8a1yTN2kFNw.

YouTube. (2021). *VulnHub - hacksudo: search. YouTube.com*. Retrieved June 28, 2023, from https://www.youtube.com/watch?v=xX9dsDBdb3A.