**Hacking Log Substitution Week 2**

One of the more fun/entertaining exploits so far was definitely the privilege escalation after obtaining the reverse shell connected to the Cuiteur web server.

Once I got shell access to Cuiteur, the feeling was amazing. It quickly dissipated, however, when I realized I couldn't execute most Linux commands. I spent a lot of time simply trying to run commands such as "sudo -l" to see what permissions I had, before realizing I had to get a better shell first. I went to the suggested readings on Canvas and found the Python command to upgrade my shell. I struggled at first, however, as python wasn't a valid command on the server, so I assumed for a good amount of time that python wasn't installed at all. I eventually figured out that the executable was called python3.6, so I was able to run the command "python3.6 -c 'import pty; pty.spawn("/bin/bash")'" to finally upgrade my shell! This was a command I must have pasted over 50 times so far, every time I accessed the shell.

With a new access to the shell, I now tried to run "sudo -l" but found out that unlike the crash course, I still needed a password to access that command. Deviating away from that approach, I decided to try and figure out which users were in the system. For this, I accessed the /etc/passwd file, and saw a large list of users. While this didn't seem to be very useful at first, it gave me a good idea of where I wanted to go later. I kept track of users printer and ubuntu which were both in the sudo group.

After several hours of trying different approaches, I finally gave up and went to get a hint. The hint talked about how files that are executed by root are executed with root privileges, so my first thought was to list all the processes that have been run on the machine. After some googling, I came across "ps -aux" to achieve this, and found a lot of processes being run by root.

Unfortunately, these processes were simply named [kworker (something)] which was extremely unhelpful. It did, however, show me that root was running processes automatically and periodically. This made me consider Cronjobs, so once again I started googling where Linux lists its cronjobs. I saw a reference to a file called “/etc/crontab”, and to my delight it was in the system! That file referenced several folders: cron.d, cron.hourly, cron.daily, cron.weekly, and cron.monthly. Inside the cron.hourly there was a file, cuiteur-cleaning, made by my user (www-data). This meant I had permissions to edit it so I knew I was in the right place. My approach was that I wanted to change the passwords of ubuntu and printer to something of my choice. Since I had to do this in one command, I did some investigating on the format of the passwd command as root. Since it asks you to confirm your password new twice, I had to pipe in the new password twice with a new line between each instance. I came up with the following command, unsure if it would work: /bin/echo -e "corner\ncorner" | /usr/bin/passwd ubuntu. Anxiously, I waited for about 30 minutes before the cron job ran, and I was able to login to ubuntu! I also decided to add my public key to ubuntu’s authorized_keys so that I wouldn’t have to create a reverse shell every time I wanted to login to it. Eventually, I used the same technique as above to change root’s password, which gave me access to the root folder with the flag inside.