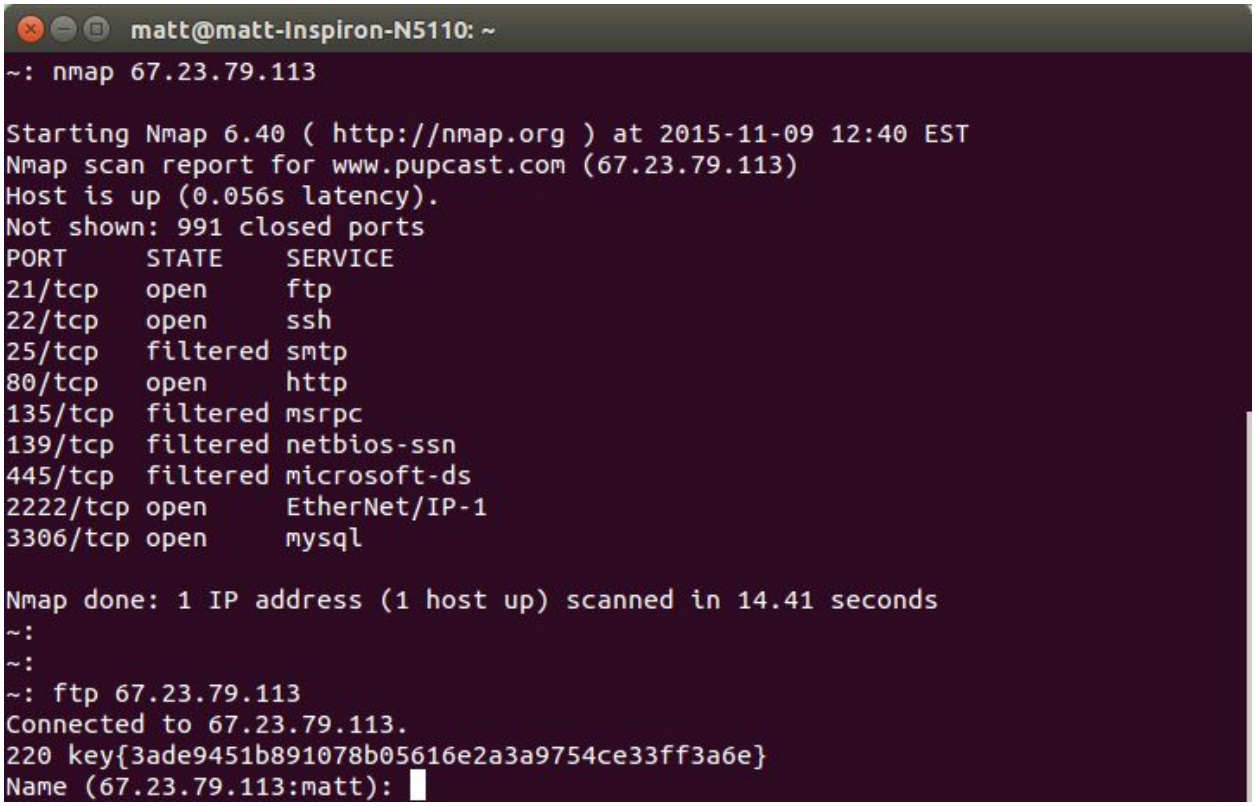
Matt Long, Walton Lee, Skyler Tom, Max Cohen
CTF Write-up for Team 7

Challenge 1: You are staring right at it



Path: nmap 67.23.79.113

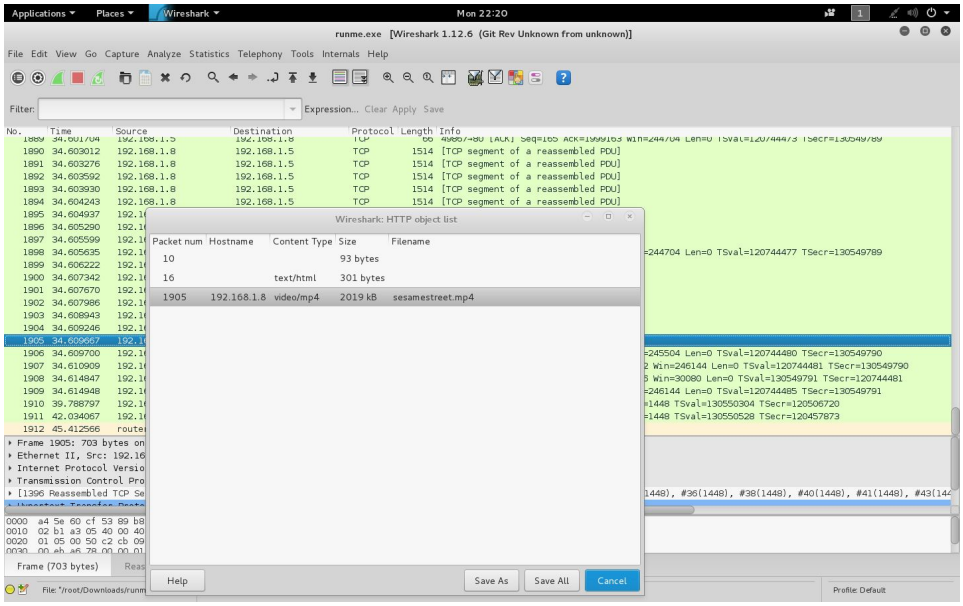
Methodology: Obviously we should try attacking the server to see if there are any blatant vulnerabilities staring us in the face. Using the nmap command above and launching an nmap attack on the ip of the capture the flag game, we located multiple open ports on the CTF server. An attempt to connect via the ftp port, the first open one, yields a message containing one of the flags.

Challenge 3: Analyze the Binary:

If we download the binary and open it up, we can see how to find the key:

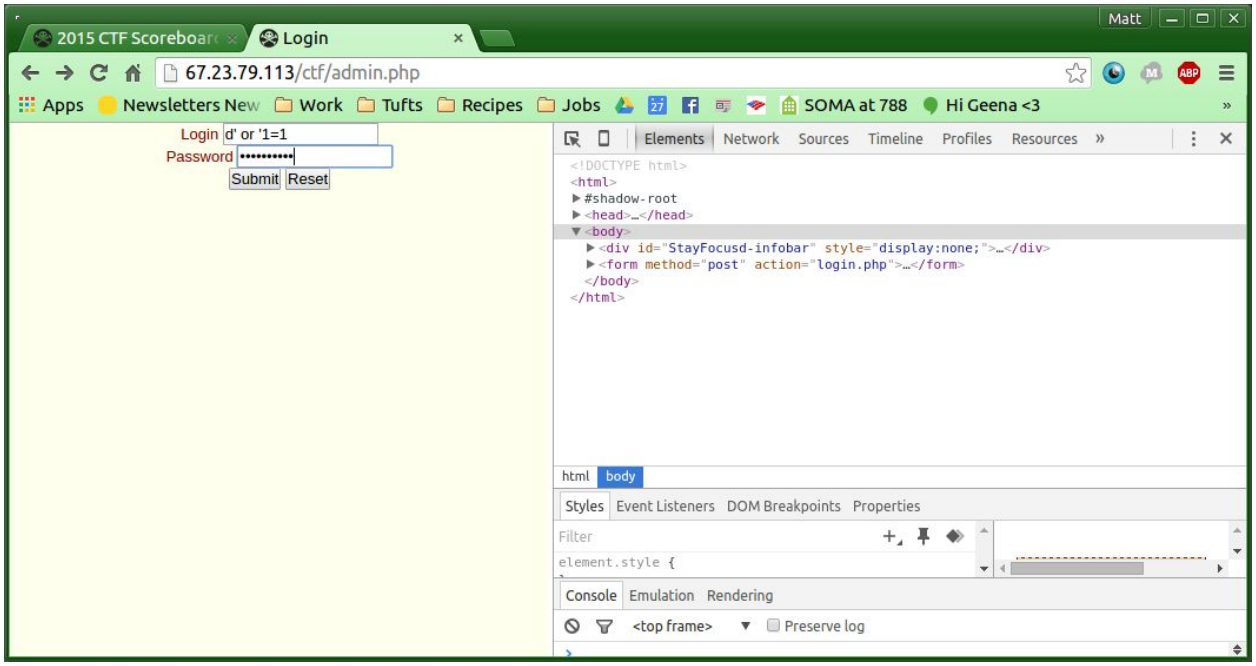
And if we look at a little more of the binary, we can see that it looks like a pcap file we can open in wireshark. We can also see that we want to watch the sesamestreet.mp4 file (the video)

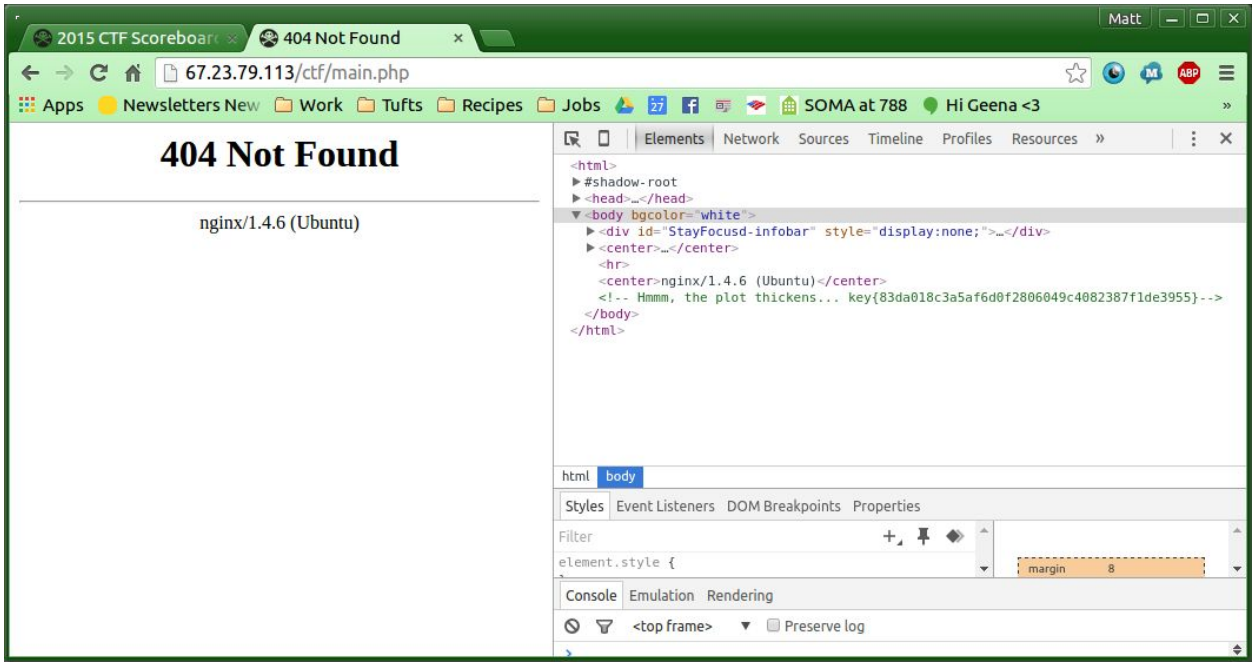
Then when we open it up in wireshark, we can go to File->Export Objects->HTTP and we can see the sesamestreet.mp4 file and download it



We watch the video, see the number 7, put it in SHA1 as SEVEN and the key is '**cabd534c35ee6a39365f4ed3bce4eafdcc3d4b8d'**

Challenge 6: Don't ask me if something looks wrong. Look again, pay careful attention:





Path: 67.23.80.113/ctf/admin.php -> /main.php

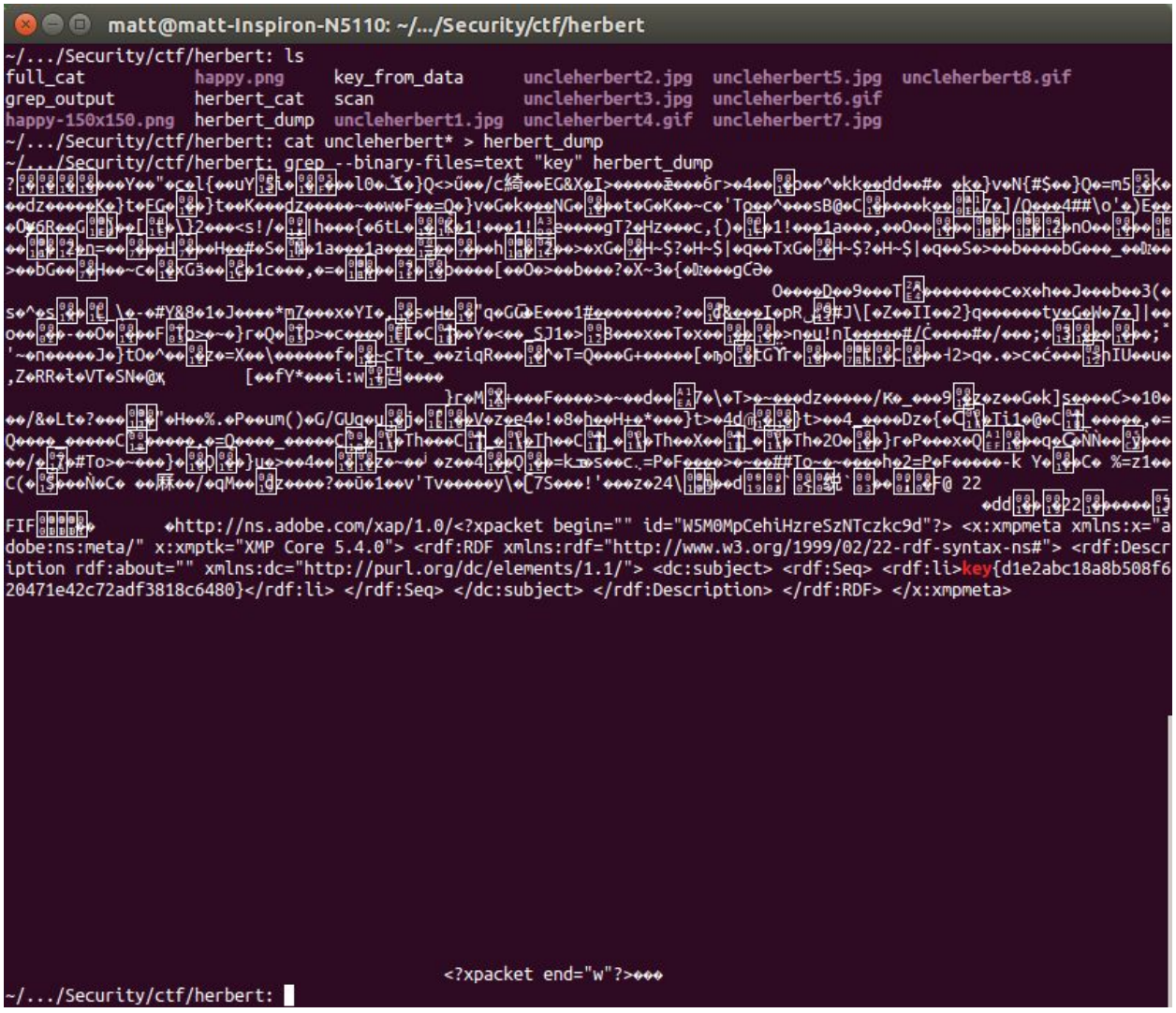
Methodology: Many WordPress sites have logins or admin pages that can be accessed by simply tacking /admin or /login onto the url. We tried /admin.php and found this site in the first screenshot. The page asked for a login and password, so naturally we tried attacking it. Attempting to perform a SQL injection attack by filling both the Login and Password fields with basic true strings such as `d' or '1=1` yields a 404 Not Found page, but investigating the source code of the 404 page yields a comment containing a flag.

Challenge 9: Buried in the dump, part 1: wide open:

Path: 67.23.79.113/ctf/wp-content/uploads/2015/10/README.txt

Methodology: In the beginning of the class, we learned about how many WordPress sites have files/pages/content out in the open under the /wp-content folder. Browsing through the easily accessible wp-content/uploads path, which is pretty standard on wordpress pages, on the capture-the-flag server gives a series of image files, along with a basic README.txt file containing one of the codes.

Challenge 10: Buried in the dump, part 1A: metadata p0rn:



Along with the README.txt, there were a bunch of pictures of Uncle Herbert from Family Guy, named uncleherbert1 - uncleherbert8. We wanted to check if any keys were hidden in those files, so we ran `cat uncleherbert* > herbert_dump` to concatenate all the uncleherbert files into one, then ran `grep --binary-files=text "key" herbert_dump` to search through the files for the string "key". The `--binary-files=text` argument ended up being unnecessary, because the key was hidden in plain text in the metadata of uncleherbert2.jpg and not within the binary data.