Chad Ballay

CYBR 430-342N

12/06/2020

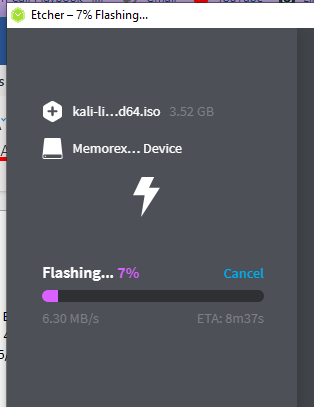
**WEEK 3 – WiFi Cracking**

**Part 1: Capture a WPA Handshake**

The handshake referred to as the WPA2 Handshake is more properly termed the 4-Way Handshake. This the initial set of communications between the device and the AP that establishes and secures the connection. This is done through an exchange of intermediary values. An extremely thorough way to understand this can be viewed here: <https://www.wifi-professionals.com/2019/01/4-way-handshake>

Suffice it to say this process is contingent on the number of times this event can be captured for analysis. To do this an attacker will attack a device’s connection to force them to deauthenticate. Doing so using Aircrack-NG required me to burn a Live CD version of Kali Linux due to not being able to manipulate the Wi-Fi card directly. (This really may be that virtualization and networking are not my strong suites.)

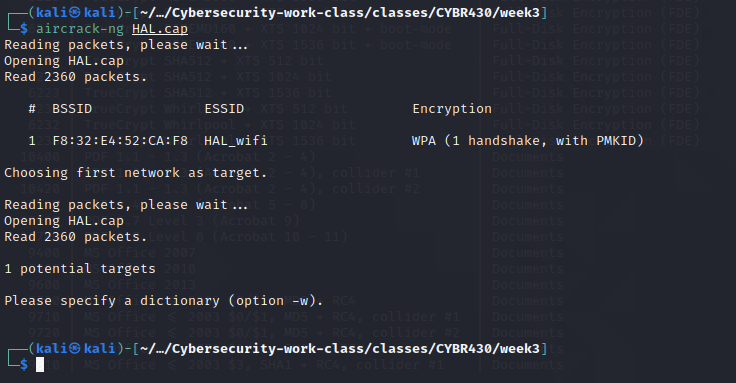
As a side note, I’m downloading and will flash USB thumb drive to boot into since I’m hesitant to try and setup a dual boot configuration and I need to keep this Windows machine working for now due to a proctored test in the next couple of weeks.



Since this was only to need to be an academic attack, I left that finish while proceeding with reading how others performed this. The general gist is that we will trigger a deauthentication to speed up this process but that is not entirely required. Instead, the core of this attack is to generate a capture of the traffic by monitoring and recording it. Any 4-way handshakes are then attacked either by brute force or by using a dictionary attack. A great overview can be found here. <http://stuffjasondoes.com/2018/07/18/capturing-wpa2-psk-handshake-with-kali-linux-and-aircrack/>

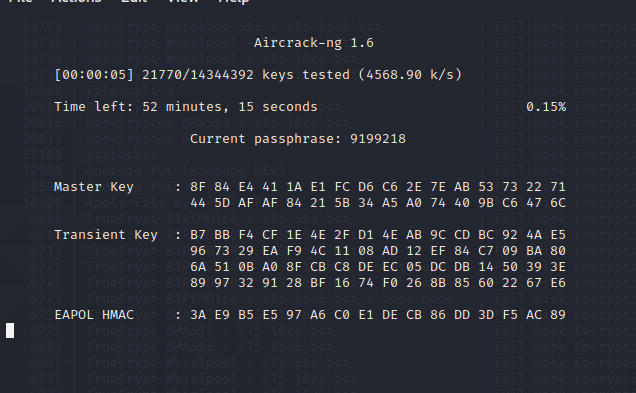
**Part 2: Cracking a captured WPA handshake**

Validated that at least 1 handshake was present. The lab had extra detail about pulling out traffic specific to one network. I also read some examples where the CAP file would be processed using WireShark or similar software to filter out what was needed.

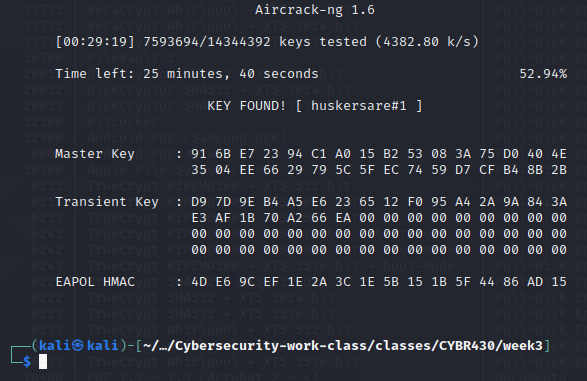


Opened up rockyou.txt file since it was compressed. (Needed to sudo over and I really need to just face the situation and create a dedicated machine.) Curiosity had me learn more about the providence of the rockyou.txt file and its history. The file itself is dated back to over a decade ago but it provides a very large dataset for a dictionary. I enjoyed this analysis that reiterates the reality of how poor password hygiene is. <https://www.passcape.com/index.php?section=blog&cmd=details&id=17>

Kicked off aircrack-ng with the rockyou dictionary. (If I were a better cloud guy I would spin up a VM in the cloud and use some GPU’s to speed this up. This kind of efficiency gains are on the TODO list.)



After working through ~50% of the keyspace the key was found!



**PSK: huskersare#1**