C15: Hashcat

Saturday, December 23, 2023 4:54 PM

Difficulty: Level 2
Eve Snowshoes is trying to recover a password. Head to the Island of Misfit Toys and take a crack at it!

CONVERSATION w/ Elf Eve Snowshoes	**Eve Snowshoes (Scaredy Kite Heights)** Greetings, fellow adventurer! Welcome to Scaredy-Kite Heights, the trailhead of the trek through the mountains on the way to the wonderful Squarewheel Yard! I'm Eve Snowshoes, resident tech hobbyist, and I hear Alabaster is in quite the predicament. Our dear Alabaster forgot his password. He's been racking his jingle bells of memory with no luck. I've been trying to handle this password recovery thing parallel to this hashcat business myself but it seems like I am missing some tricks. So, what do you say, chief, ready to get your hands on some hashcat action and help a distraught elf out? ** ------- Response after completing challenge ------- ** Aha! Success! Alabaster will undoubtedly be grateful for our assistance. Onward to our next adventure, comrade! Feel free to explore this whimsical world of gears and steam! From <https://2023.holidayhackchallenge.com/badge?section=conversation&id=evesnowshoes>
HINTS	

Its type and form, in whispers, spoken.
From reindeers' leaps to the elfish toast,
Might the secret be in an ASREP roast?

`hashcat`, your reindeer, so spry and true,
Will leap through hashes, bringing answers to you.
But heed this advice to temper your pace,
`-w 1 -u 1 --kernel-accel 1 --kernel-loops 1`, just in case.

For within this quest, speed isn't the key,
Patience and thought will set the answers free.
So include these flags, let your command be slow,
And watch as the right solutions begin to show.

For hints on the hash, when you feel quite adrift,
This festive link, your spirits, will lift:
https://hashcat.net/wiki/doku.php?id=example_hashes

And when in doubt of `hashcat`'s might,
The CLI docs will guide you right:
https://hashcat.net/wiki/doku.php?id=hashcat

Once you've cracked it, with joy and glee so raw,
Run /bin/runtoanswer, without a flaw.
Submit the password for Alabaster Snowball,
Only then can you claim the prize, the best of all.

So light up your terminal, with commands so grand,
Crack the code, with `hashcat` in hand!
Merry Cracking to each, by the pixelated moon's light,
May your hashes be merry, and your codes so right!

* Determine the hash type in hash.txt and perform a wordlist cracking attempt to find which password is correct and submit it to /bin/runtoanswer .*

MY WORK AND ANSWER

I enjoyed this challenge. The poem provided plenty of hints and I used MS AI to help with identifying and deciphering comman d line syntax.

<table>
<tr><td>Decipher hints in poem</td><td>ASREP has code of 18200<table><tr><td>18200</td><td>Kerberos</td><td>$krb5asrep$23$user@domain.com:3e156ada591263b8aab0965f5aebd837</td></tr></table></td></tr>
</table>

	5, etype 23, AS-REP	$007497cb51b6c8116d6407a782ea0e1c5402b17db7afa6b05a6d30ed164a9933c754d720e279c6c573679bd27128fe77e5fea1f72334c1193c8ff0b370fadc6368bf2d49bbfdba4c5dccab95e8c8ebfdc75f438a0797dbfb2f8a1a5f4c423f9bfc1fea483342a11bd56a216f4d5158ccc4b224b52894fadfba3957dfe4b6b8f5f9f9fe422811a314768673e0c924340b8ccb84775ce9defaa3baa0910b676ad0036d13032b0dd94e3b13903cc738a7b6d00b0b3c210d1f972a6c7cae9bd3c959acf7565be528fc179118f28c679f6deeee1456f0781eb8154e18e49cb27b64bf74cd7112a0ebae2102ac

From <https://hashcat.net/wiki/doku.php?id=example_hashes>

Check directory and files in local directory. - Found the hash file and the password cracking list	elf@273d149932a9:~$ pwd /home/elf elf@273d149932a9:~$ ls -la total 40 drwxr-xr-x 1 elf elf 4096 Nov 27 17:07 . drwxr-xr-x 1 root root 4096 Nov 20 18:07 .. -rw-r--r-- 1 elf elf 220 Feb 25 2020 .bash_logout -rw-r--r-- 1 elf elf 3771 Feb 25 2020 .bashrc -rw-r--r-- 1 elf elf 807 Feb 25 2020 .profile -rw-r--r-- 1 elf elf 1567 Nov 27 17:07 HELP -rw-r--r-- 1 elf elf 541 Nov 9 21:29 hash.txt -rw-r--r-- 1 root root 2775 Nov 9 21:29 password_list.txt
The hash is a Kerberos	elf@273d149932a9:~$ cat hash.txt $krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879eda02$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e48e019256e466d29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd2772b908fc0d0aef43bb34db66af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479bae25c0183e74822ac652e92a56d0251bb5d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6e52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4b415574d7132f2607e12075f7cd2f8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f4b78ddf6830ad0e9eafb07980d7f2e270d8dd1966elf@273d149932a9:~$
Break the password using hashcat hashcat options hashfile pwdfile	hashcat -m 18200 -w 1 -u 1 --kernel-accel 1 --kernel-loops 1 -a0 --force hash.txt password_list.txt ... Approaching final keyspace - workload adjusted. Started: Mon Jan 1 04:12:08 2024 Stopped: Mon Jan 1 04:12:26 2024 $krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879eda02$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e48e019256e466d29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd2772b908fc0d0aef43bb34db66af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479bae25c0183e74822ac652e92a56d0251bb5d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6e52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4b415574d7132f2607e12075f7cd2f8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f4b78ddf6830ad0e9eafb07980d7f2e270d8dd1966:IluvC4ndyC4nes! Session..........: hashcat Status...........: Cracked Hash.Type........: Kerberos 5 AS-REP etype 23 Hash.Target......: $krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2...dd1966 Time.Started.....: Mon Jan 1 04:41:44 2024 (0 secs) Time.Estimated...: Mon Jan 1 04:41:44 2024 (0 secs) Guess.Base.......: File (password_list.txt) Guess.Queue......: 1/1 (100.00%) Speed.#1.........: 1218 H/s (0.67ms) @ Accel:1 Loops:1 Thr:64 Vec:16 Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts Progress.........: 144/144 (100.00%) Rejected.........: 0/144 (0.00%) Restore.Point....: 0/144 (0.00%) Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-0 Candidates.#1....: 1LuvCandyC4n3s!2022 -> iLuvC4ndyC4n3s!23! Started: Mon Jan 1 04:41:28 2024 Stopped: Mon Jan 1 04:41:46 2024 elf@f7c1c7973e7c:~$
Directory listing after running hashcat	elf@273d149932a9:/home$ cd elf elf@273d149932a9:~$ ls -la total 44 drwxr-xr-x 1 elf elf 4096 Jan 1 04:12 .

```
drwxr-xr-x 1 root root 4096 Nov 20 18:07 ..
-rw-r--r-- 1 elf  elf   220 Feb 25  2020 .bash_logout
-rw-r--r-- 1 elf  elf  3771 Feb 25  2020 .bashrc
drwx------ 3 elf  elf  4096 Jan  1 04:12 .cache
drwx------ 4 elf  elf  4096 Jan  1 04:12 .hashcat
-rw-r--r-- 1 elf  elf   807 Feb 25  2020 .profile
-rw-r--r-- 1 elf  elf  1567 Nov 27 17:07 HELP
-rw-r--r-- 1 elf  elf   541 Nov  9 21:29 hash.txt
-rw-r--r-- 1 root root 2775 Nov  9 21:29 password_list.txt
```

```
elf@f7c1c7973e7c:~/.hashcat$ cd ..
elf@f7c1c7973e7c:~$ ls -la
total 44
drwxr-xr-x 1 elf  elf  4096 Jan  1 04:41 .
drwxr-xr-x 1 root root 4096 Nov 20 18:07 ..
-rw-r--r-- 1 elf  elf   220 Feb 25  2020 .bash_logout
-rw-r--r-- 1 elf  elf  3771 Feb 25  2020 .bashrc
drwx------ 3 elf  elf  4096 Jan  1 04:41 .cache
drwx------ 4 elf  elf  4096 Jan  1 04:41 .hashcat
-rw-r--r-- 1 elf  elf   807 Feb 25  2020 .profile
-rw-r--r-- 1 elf  elf  1567 Nov 27 17:07 HELP
-rw-r--r-- 1 elf  elf   541 Nov  9 21:29 hash.txt
-rw-r--r-- 1 root root 2775 Nov  9 21:29 password_list.txt
elf@f7c1c7973e7c:~$ cd .hashcat
elf@f7c1c7973e7c:~/.hashcat$ ls -la
total 24
drwx------ 4 elf elf 4096 Jan  1 04:41 .
drwxr-xr-x 1 elf elf 4096 Jan  1 04:41 ..
-rw------- 1 elf elf  296 Jan  1 04:41 hashcat.dictstat2
-rw------- 1 elf elf  558 Jan  1 04:41 hashcat.potfile
drwx------ 2 elf elf 4096 Jan  1 04:41 kernels
drwx------ 2 elf elf 4096 Jan  1 04:41 sessions

elf@f7c1c7973e7c:~/.hashcat$ cat hashcat.potfile
$krb5asrep$23$alabaster_snowball@XMAS.LOCAL:22865a2bceeaa73227ea4021879eda02
$8f07417379e610e2dcb0621462fec3675bb5a850aba31837d541e50c622dc5faee60e48e019256e466d
29b4d8c43cbf5bf7264b12c21737499cfcb73d95a903005a6ab6d9689ddd2772b908fc0d0aef43bb34db6
6af1dddb55b64937d3c7d7e93a91a7f303fef96e17d7f5479bae25c0183e74822ac652e92a56d0251bb5
d975c2f2b63f4458526824f2c3dc1f1fcbacb2f6e52022ba6e6b401660b43b5070409cac0cc6223a2bf1b4
b415574d7132f2607e12075f7cd2f8674c33e40d8ed55628f1c3eb08dbb8845b0f3bae708784c805b9a3f
4b78ddf6830ad0e9eafb07980d7f2e270d8dd1966:IluvC4ndyC4nes!
```

IluvC4ndyC4nes!

```
elf@f7c1c7973e7c:/run$ runtoanswer
What is the password for the hash in /home/elf/hash.txt ?

> IluvC4ndyC4nes!
Your answer: IluvC4ndyC4nes!

Checking....
Your answer is correct!
```

Comments of Interest from other Players

Thats legit. Solve it however works best. But yea the intent is to use the tuning arguments in HELP so it can run in a limite d memory env.
discover the hash type, reading the hint is important**and know the basic hashcat formula :)

The error message I got said use the --force option. This allows use of outdated flags in the command

The candidates part of the hashcat output is just a way of outputing the first and last entry in the wordlist or wordlist+rul es. It has nothing to do with a solution, it's just for info so you know it read your intentions right

Not a copy paste error.
There is a third candidate in there somewhere

Hashcat stores the results in a potfile yes. The easiest way to view the results is to run the same command you used to crack the hashes and add --show to it
On the HHC terminal, the path is ~/.hashcat/hashcat.potfile

[12:47 PM]John_r2: You can cat the file, or as @FluffMe suggested, use the --show option from hashcat.

The cracked password is a bit higher up in the output. Alternatively, like some players already comment, you can run the prog ram with —show after successful cracking

A feature/problem of JohnTheRipper and Hashcat is they keep a list of hashes they've already cracked. That way they don't wa ste time cracking the same hash on repeated runs (feature). If you miss that an early run cracked the hash, then you can run hashcat a zillion tim es and never see that hash cracked (problem). They store the previous hashes in a potfile--the name pot dates to the 90's https://www.openwall.com/lists/john-users/2015/09/10/4.

The real cracked password can be viewed by adding --show to the command. It will show the entry it cracked from the potfile

hint to trying to identify the hash encyrption type. || look at the beginning of the hash the parts starting with $ ||.
i used hashcat -m18200 -a0 hash.txt password_list.txt --force

cd /bin then ./runtoanswer

I found another linear dependency. I solved the hashcat challenge and the objective did not show up until I went to the Black light District.

This one was especially helpful: https://www.youtube.com/watch?v=R_Nsj6BUr6w&t=8s

----

Brute-forcing secret keys using hashcat

We recommend using hashcat to brute-force secret keys. You can install hashcat manually, but it also comes pre-installed and ready to use on Kali Linux.

Note

If you're using the pre-built VirtualBox image for Kali rather than the bare metal installer version, this may not have enough memory allocated to ru n hashcat.

You just need a valid, signed JWT from the target server and a wordlist of well-known secrets. You can then run the following command, passing in the JWT and wordlist as arguments:

```
hashcat -a 0 -m 16500 <jwt> <wordlist>
```

Hashcat signs the header and payload from the JWT using each secret in the wordlist, then compares the resulting signature wi th the original one from the server. If any of the signatures match, hashcat outputs the identified secret in the following format, along with various other details:

```
<jwt>:<identified-secret>
```

Note

If you run the command more than once, you need to include the --show flag to output the results.

As hashcat runs locally on your machine and doesn't rely on sending requests to the server, this process is extremely quick, even when using a huge wordlist.

Once you have identified the secret key, you can use it to generate a valid signature for any JWT header and payload that you like. For details on how to re-sign a modified JWT in Burp Suite, see Editing JWTs.

From <https://portswigger.net/web-security/jwt>