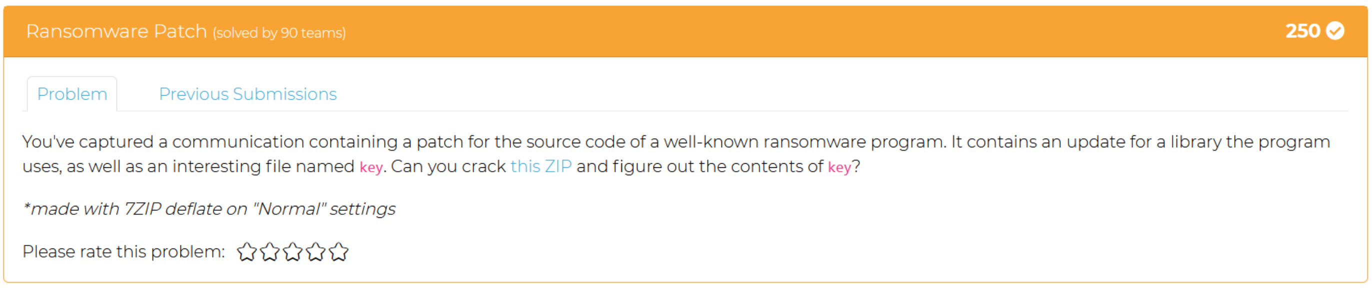
MetaCTF 2021 Writeup: Ransomware Patch



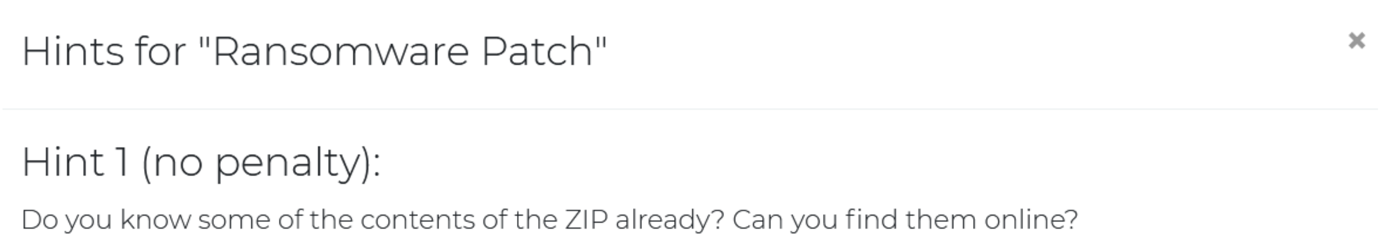
Ransomware Patch (solved by 90 teams) 250

Problem Previous Submissions

You've captured a communication containing a patch for the source code of a well-known ransomware program. It contains an update for a library the program uses, as well as an interesting file named key. Can you crack this ZIP and figure out the contents of key?

*made with 7ZIP deflate on "Normal" settings

Please rate this problem: ☆☆☆☆☆



Hints for "Ransomware Patch"

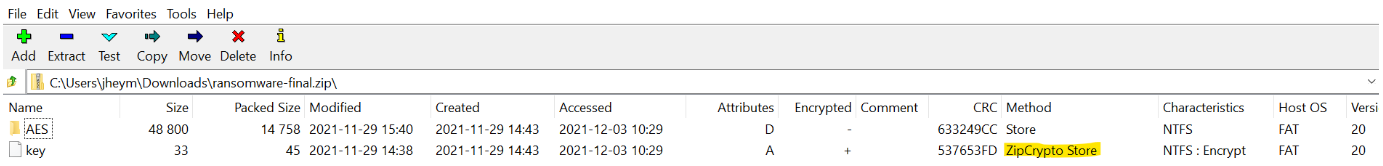
Hint 1 (no penalty):

Do you know some of the contents of the ZIP already? Can you find them online?

[Download Link for ZIP file]()

Background

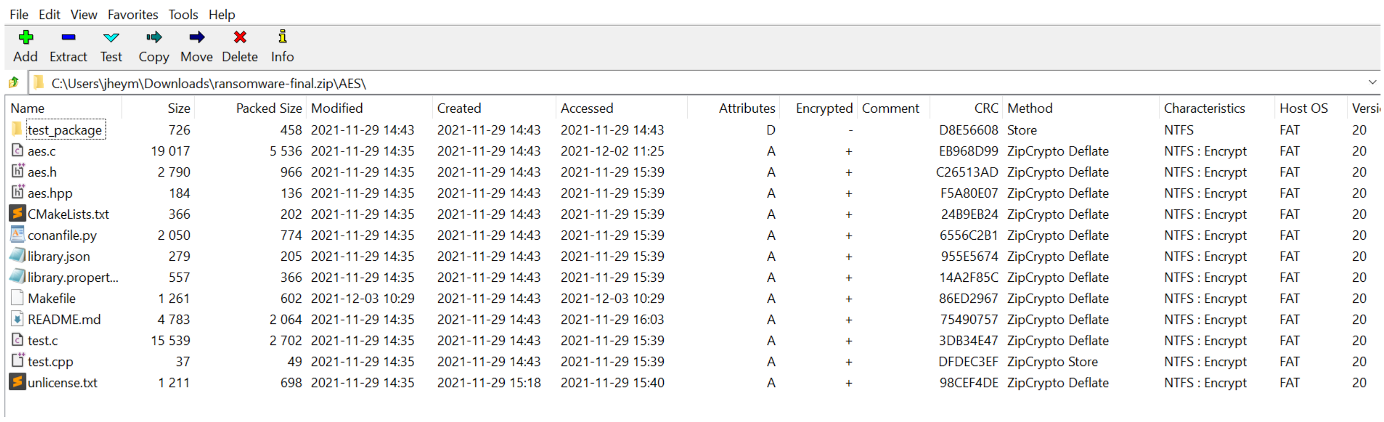
When I first saw this challenge I thought it would just be as simple as using zip2john and running rockyou against it. Boy was I wrong. I tried using John the Ripper and fcrackzip and they didn't seem to know how to handle the way 7zip encrypts files, as opposed to using winzip or pkzip. As the challenge mentions, the zip archive was made in 7zip. So I opened the archive in 7zip to try to see if there was any useful information about how the archive was created.


File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info

C:\Users\jheym\Downloads\ransomware-final.zip\

Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Characteristics	Host OS	Versi
AES	48 800	14 758	2021-11-29 15:40	2021-11-29 14:43	2021-12-03 10:29	D	-		633249CC	Store	NTFS	FAT	20
key	33	45	2021-11-29 14:38	2021-11-29 14:43	2021-12-03 10:29	A	+		537653FD	ZipCrypto Store	NTFS : Encrypt	FAT	20

It looks like the encryption method used is called "ZipCrypto Store." A google search of the term results in various articles stating that this is a legacy encryption method prone to known plaintext attacks. That is, if you know at least 12 bytes of data in any of the encrypted files, you can find the encryption keys within minutes. I needed to guess some plaintext that I know will be contained in the archive. Thankfully, we can see the names of all of the files in the archive and the file size before compression.


File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info

C:\Users\jheym\Downloads\ransomware-final.zip\AES\

Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Characteristics	Host OS	Versi
test_package	726	458	2021-11-29 14:43	2021-11-29 14:43	2021-11-29 14:43	D	-		D8E56608	Store	NTFS	FAT	20
aes.c	19 017	5 536	2021-11-29 14:35	2021-11-29 14:43	2021-12-02 11:25	A	+		EB968D99	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
aes.h	2 790	966	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		C26513AD	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
aes.hpp	184	136	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		F5A80E07	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
CMakeLists.txt	366	202	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		24B9EB24	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
conanfile.py	2 050	774	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		6556C2B1	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
library.json	279	205	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		955E5674	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
library.propert...	557	366	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		14A2F85C	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
Makefile	1 261	602	2021-12-03 10:29	2021-11-29 14:43	2021-12-03 10:29	A	+		86ED2967	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
README.md	4 783	2 064	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 16:03	A	+		75490757	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
test.c	15 539	2 702	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		3DB34E47	ZipCrypto Deflate	NTFS : Encrypt	FAT	20
test.cpp	37	49	2021-11-29 14:35	2021-11-29 14:43	2021-11-29 15:39	A	+		DFDEC3EF	ZipCrypto Store	NTFS : Encrypt	FAT	20
unlicense.txt	1 211	698	2021-11-29 14:35	2021-11-29 15:18	2021-11-29 15:40	A	+		98CEF4DE	ZipCrypto Deflate	NTFS : Encrypt	FAT	20

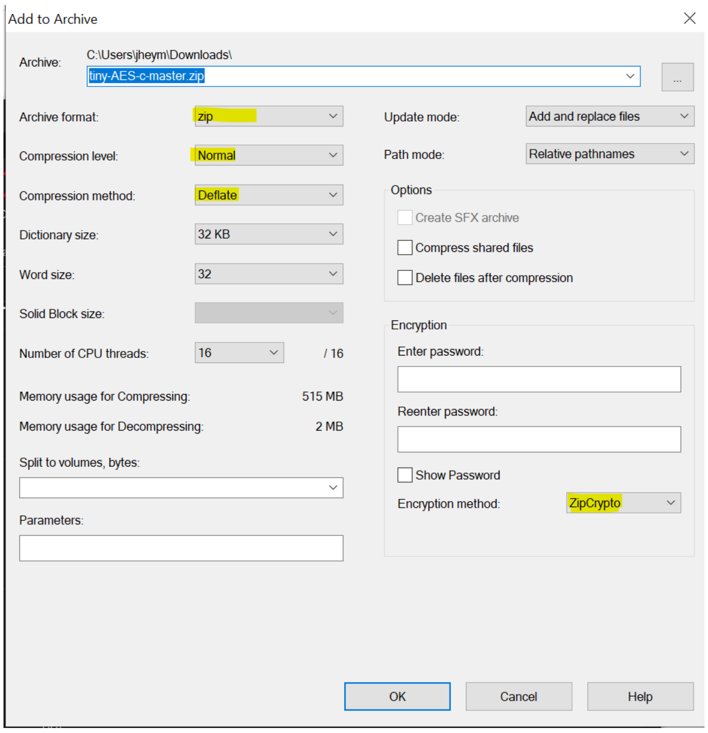
I needed guess at some plaintext that was definitely encrypted in there. My first thought was to use the unlicense.txt as a plaintext, as it is a pretty common license that people just copy word-for-word. But it didn't work for me. I also noticed that the file size of my unlicense.txt was different than that of the file size noted in the archive. So I went looking up some of the other files online and comparing file sizes. I found an aes.hpp that was the exact same file size (184 bytes) from a Github repo. That didn't seem to work for me either. I must not have been setting it up right.

After a day of doing other challenges, I wanted to give this another go because I knew it was close. I didn't realize it before, but that github repo with the .hpp file was an exact match to the contents in the ransomware archive (shoutout to Isaac). The comp was about to end in 15 minutes so I had to be quick.

Solution

1. For bkcrack, we need to make an unencrypted version of the plaintext using the same compression method used to make the encrypted archive. This information was given in the challenge "Made with 7ZIP deflate on normal settings." I downloaded and unzip the folder from the github repo containing the plaintext matches. https://github.com/kokke/tiny-AES-c
2. Now open the folder in 7zip and create the zip archive using deflate and normal settings. Leave password encryption blank, but notice how zipcrypto is actually the default 7ZIP encryption

method? Funny, right?


Add to Archive

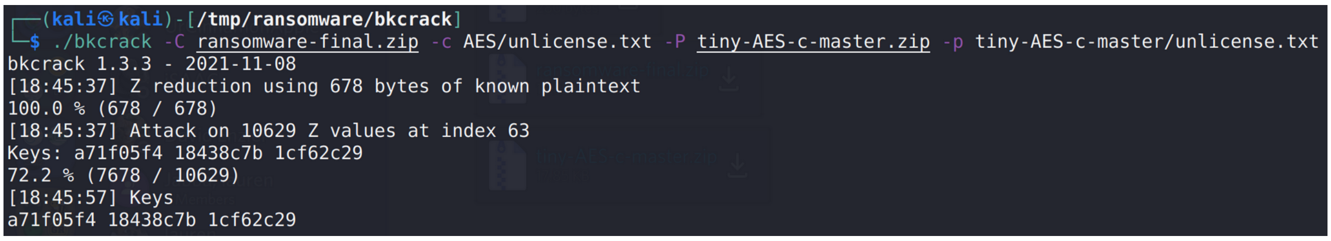
Archive: C:\Users\jheym\Downloads\
tiny-AES-c-master.zip

Archive format: zip
Compression level: Normal
Compression method: Deflate
Dictionary size: 32 KB
Word size: 32
Solid Block size:
Number of CPU threads: 16 / 16
Memory usage for Compressing: 515 MB
Memory usage for Decompressing: 2 MB
Split to volumes, bytes:
Parameters:

Update mode: Add and replace files
Path mode: Relative pathnames
Options
Create SFX archive
Compress shared files
Delete files after compression
Encryption
Enter password:
Reenter password:
Show Password
Encryption method: ZipCrypto

3. I moved my new plaintext archive into Kali to use with bkcrack tool, which uses Biham and Kocher's known plaintext attack. There are a couple methods for this tool, but the one that worked for me was the one that uses two zip files.
4. Move the two zip files into the bkcrack folder that contains the executable and execute this command:
`./bkcrack -C ransomware-final.zip -c AES/unlicense.txt -P tiny-AES-c-master.zip -p tiny-AES-c-master/unlicense.txt`

- `-C encrypted archive`
- `-c cipher file (name of encrypted file in the encrypted archive)`
- `-P plaintext unencrypted archive we created using same compression method`
- `-p plaintext file in the archive we want to use`

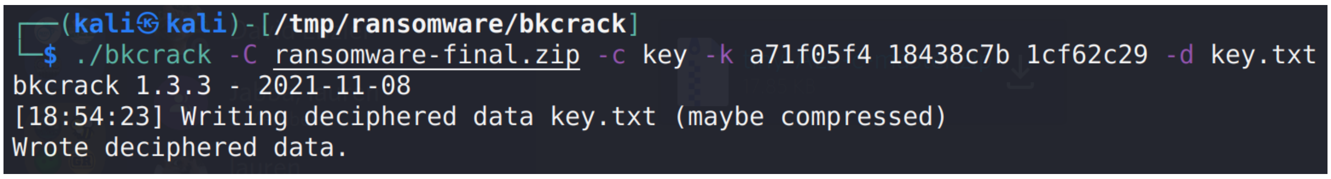

```
┌──(kali㉿kali)-[/tmp/ransomware/bkcrack]
└─$ ./bkcrack -C ransomware-final.zip -c AES/unlicense.txt -P tiny-AES-c-master.zip -p tiny-AES-c-master/unlicense.txt
bkcrack 1.3.3 - 2021-11-08
[18:45:37] Z reduction using 678 bytes of known plaintext
100.0 % (678 / 678)
[18:45:37] Attack on 10629 Z values at index 63
Keys: a71f05f4 18438c7b 1cf62c29
72.2 % (7678 / 10629)
[18:45:57] Keys
a71f05f4 18438c7b 1cf62c29
```

Using 678 bytes of known plaintext takes less than a minute to crack. Once we have the keys, we can open other encrypted files.

5. `./bkcrack -C ransomware-final.zip -c key -k a71f05f4 18438c7b 1cf62c29 -d key.txt`

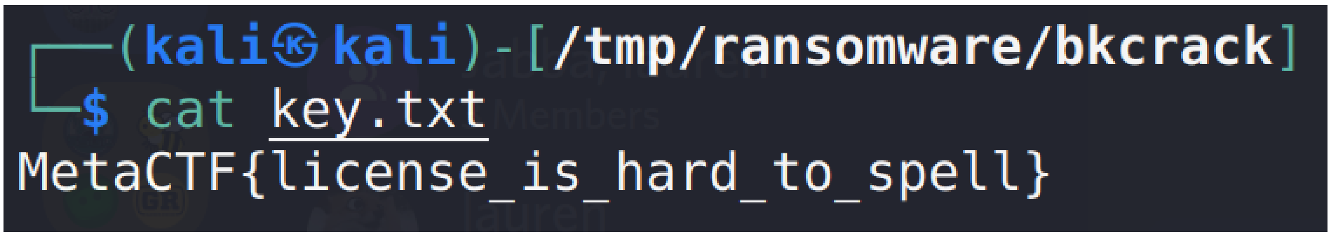
- `-C encrypted zip archive`

- `-c specified file within the encrypted archive that we want to use the key on to decrypt`
- `-k the keys we got from the last step (replace mine with yours)`
- `-d the name of file you want to put the decrypted data into`



```
┌──(kali㉿kali)-[/tmp/ransomware/bkcrack]
└─$ ./bkcrack -C ransomware-final.zip -c key -k a71f05f4 18438c7b 1cf62c29 -d key.txt
bkcrack 1.3.3 - 2021-11-08
[18:54:23] Writing deciphered data key.txt (maybe compressed)
Wrote deciphered data.
```

6. See what's in the decypted file key.txt



```
┌──(kali㉿kali)-[/tmp/ransomware/bkcrack]
└─$ cat key.txt
MetaCTF{license_is_hard_to_spell}
```

It works!!!

Thanks for reading - Justin

References

[Why you should never use zipcrypto]()

[Kai Anter - How to do a ZipCrypto plaintext attack]()

https://github.com/kimci86/bkcrack/blob/master/readme.md

https://superuser.com/a/859930