Write-up for

NC3 Jule-CTF 2021

opgave "bob"

Indledning
Opgaven består er en 512 byte blok. Nedenfor ses et hexdump af hele blokken.

```
--------------------------------------------------------------------------
   0   0D 00 00 00 17 00 68 00  01 F3 01 E4 01 D1 01 C5   ......h.........
  10   01 B1 01 99 01 8B 01 72  01 5B 01 4A 01 2E 01 17   .......r.[.J....
  20   01 06 00 F7 00 E7 00 DB  00 CC 00 BB 00 AB 00 9A   ................
  30   00 8D 00 75 00 68 00 00  00 00 00 00 00 00 00 00   ...u.h..........
  40   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ................
  50   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ................
  60   00 00 00 00 00 00 00 00  08 86 8E F0 76 03 00 17   ............v...
  70   49 20 73 69 74 13 86 8E  CA 72 03 00 2D 68 65 61   I sit....r..-hea
  80   64 69 6E 67 20 73 74 72  61 69 67 68 74 08 85 99   ding straight...
  90   D6 6B 03 00 17 67 68 6F  73 74 0C 85 95 C8 65 03   .k...ghost....e.
  a0   00 1F 69 74 27 73 20 6A  75 73 74 0B 83 ED E0 72   ..it's just....r
  b0   03 00 1D 61 6E 64 20 68  65 72 65 0C 83 CD E8 6E   ...and here....n
  c0   03 00 1F 74 65 6C 65 70  68 6F 6E 65 0A 83 C8 C2   ...telephone....
  d0   20 03 00 1B 61 6E 64 20  79 6F 75 07 83 BB 87 26    ...and you....&
  e0   03 00 15 6D 6F 6F 6E 0B  83 B8 C0 6B 03 00 1D 74   ...moon....k...t
  f0   68 61 74 20 74 68 65 0A  83 AD D6 65 03 00 1B 69   hat the....e...i
 100   73 20 66 75 6C 6C 0C 83  A5 C2 6D 03 00 1F 49 27   s full....m...I'
 110   64 20 6B 6E 6F 77 6E 12  83 9D DD 43 03 00 2B 68   d known....C..+h
 120   65 72 65 20 63 6F 6D 65  73 20 79 6F 75 72 17 83   ere comes your..
 130   95 E4 2C 03 00 35 61 67  61 69 6E 20 62 75 74 20   ..,..5again but
 140   74 68 61 74 27 73 20 6E  6F 74 0C 83 95 E4 20 03   that's not.... .
 150   00 1F 79 65 61 72 73 20  61 67 6F 12 83 94 C0 64   ..years ago....d
 160   03 00 2B 68 65 61 72 69  6E 67 20 61 20 76 6F 69   ..+hearing a voi
 170   63 65 14 83 85 DC 74 03  00 2F 61 20 63 6F 75 70   ce....t../a coup
 180   6C 65 20 6F 66 20 6C 69  67 68 74 09 83 85 DC 20   le of light....
 190   03 00 19 64 61 6D 6E 65  64 13 82 B9 86 33 03 00   ...damned....3..
 1a0   2D 68 61 70 70 65 6E 65  64 20 74 6F 20 63 61 6C   -happened to cal
 1b0   6C 0F 82 91 F2 6C 03 00  25 77 65 6C 6C 20 49 27   l....l..%well I'
 1c0   6C 6C 20 62 65 07 81 DC  EA 7D 03 00 15 66 61 6C   ll be....}...fal
 1d0   6C 0E 81 81 E4 75 03 00  23 68 61 6E 64 20 6F 6E   l....u..#hand on
 1e0   20 74 68 65 0A 81 81 CF  43 03 00 1B 75 6E 75 73    the....C...unus
 1f0   75 61 6C 08 81 80 E2 39  03 00 17 66 6F 72 20 61   ual....9...for a
--------------------------------------------------------------------------
```

Blokken starter med en eller anden form for header (0x0 - 0x35) og i slutningen ligger nogle tilsyneladende systematiske data af både ASCII og binært (0x68 - 0x1FF).

Dette minder om en database og formatet passer med en såkaldt "data page" fra en SQLite-database. Det er et uddrag af en database, der mangler en "header-page" og en del meta-data. (Det er der nok nogen der har gjort med vilje.) Formatet for SQLite-databaser er veldefineret og findes på sqlite.org:

`https://www.sqlite.org/mostdeployed.html`

Analyse af rækker
På trods af manglende meta-data, kan rækkerne stadig genkendes, et uddrag ses nedenfor. Page'n består af en header, et sæt pointere til rækkerne samt rækker/celledata.

```
------------------------------------------------------------------------
  0   0D 00 00 00 17 00 68 00  01 F3 01 E4 01 D1 01 C5   ......h.........
 10   01 B1 01 99 01 8B 01 72  01 5B 01 4A 01 2E 01 17   .......r.[.J....
 20   01 06 00 F7 00 E7 00 DB  00 CC 00 BB 00 AB 00 9A   ................
 30   00 8D 00 75 00 68 00 00  00 00 00 00 00 00 00 00   ...u.h..........
...
 60   00 00 00 00 00 00 00 00  08 86 8E F0 76 03 00 17   ............v...
 70   49 20 73 69 74 13 86 8E  CA 72 03 00 2D 68 65 61   I sit....r..-hea
------------------------------------------------------------------------
```

Nedenfor ses et udvalg af rækkerne (dem med kun et ord i payload). Ved nærmere undersøgelse af den første række ses, at der umiddelbart er 2 kolonner.:

```
Offset  Bytes                                                    Ascii
------------------------------------------------------------------------------
 8d   08 85 99 D6 6B 03 00 17 67 68 6F 73 74                    ....k...ghost
 bb   0C 83 CD E8 6E 03 00 1F 74 65 6C 65 70 68 6F 6E 65        ....n...telephone
 db   07 83 BB 87 26 03 00 15 6D 6F 6F 6E                       ....&...moon
18b   09 83 85 DC 20 03 00 19 64 61 6D 6E 65 64                 .... ...damned
1c5   07 81 DC EA 7D 03 00 15 66 61 6C 6C                       ....}...fall
1e4   0A 81 81 CF 43 03 00 1B 75 6E 75 73 75 61 6C              ....C...unusual
      ... ...
------------------------------------------------------------------------------
```

Grundlæggende består hver række af 4 dele: en såkaldt *varint* med længden af payload, en varint som er ROWID, et antal varints med "serial types" og til sidst de egentlige data.

Udtræk af data
De såkaldte "varints" er SQLite komprimerede integers, som skal dekodes efter en særlig procedure, før man kan se, hvilke tal de repræsenterer. (Det er der nok nogen der har gjort med vilje.)

Hovedreglen er, at så længe den øverste bit er sat, kommer der en byte mere. Nedenfor er et eksempel på hvordan det kan gøre manuelt med de to første rækker. (Nærmere detaljer om varints findet på SQLites hjemmeside.)

```
86 8E F0 76  ->   10000110 10001110 11110000 01110110  ->   0xC3B876
86 8E CA 72  ->   10000110 10001110 11001010 01110010  ->   0xC3A572
```

Nu kan row-id og tekst trækkes ud fra rækkerne, hvilket ser således ud:

```
--------------------------
C3B876 - I sit
C3A572 - heading straight
A66B6B - ghost
A56465 - it's just
7B7072 - and here
73746E - telephone
722120 - and you
6EC3A6 - moon
```

```
6E206B - that the
6B6B65 - is full
69616D - I'd known
676EC3 - here comes your
65722C - again but that's not
657220 - years ago
652064 - hearing a voice
616E74 - a couple of light
616E20 - damned
4E4333 - happened to call
44796C - well I'll be
37357D - fall
207275 - hand on the
2067C3 - unusual
203139 - for a
-------------------------
```

I forbindelse med CTF'en ses gerne et sæt tuborg-klammer og det bemærkes at ASCII-koden for { og }, henholdsvis 0x7B og ox7D findes i den 5. og den 20. linie. Linie 18 indholder 0x4E, 0x43 og 0x33, hvilket jo sjovt nok bliver til "NC3".

Der er dog også mange hex-koder i tallene, der ligger udenfor 7-bit ASCII. Det kunne tænkes at være UTF-8. (Det er der nok nogen, der har gjort med vilje...) Måske er det "bare" er et spørgsmål om at sortere i den rigtige rækkefølge.

Det sidste spor

Af de ialt 23 fremkomne linier, indeholder 6 af dem kun et enkelt ord, som til gengæld er ret karakteristisk (ses ovenfor). Hvis disse 6 ord googles (eller lign.) finder man "Diamonds and Rust" af Joan Baez. Ovenstående linier kan sorteres, så de danner første vers af teksten til denne sang.

Showdown

Sættes alle row-ids i korrekt orden efter det først vers i sangen og dekodes som UTF-8, dukker beskeden op. Her er det gjort med python:

```
>>> bytes.fromhex("44 79 6C 61 6E 20 67 6E C3 A6 6B 6B 65 72 2C 20 67 C3 A5 64
65 6E 20 6B 6E C3 A6 6B 6B 65 72 21 20 4E 43 33 7B 70 72 C3 B8 76 20 72 75 73 74
6E 65 20 64 69 61 6D 61 6E 74 65 72 20 C3 A5 72 20 31 39 37 35
7D") .decode("utf-8")
'Dylan gnækker, gåden knækker! NC3{prøv rustne diamanter år 1975}'
>>>
```

Bonusinfo

Sangen er skrevet i 1975 og handler om Bob Dylan. Hvis man google "bob" med noget af teksten, får man også meget hurtigt fundet det rigtige vers frem.