Solve

Rainbow Table Creation

It takes 28 minutes to generate a rainbow table corresponding to 50 billion hashes.

Of course there are a lot of duplicates in this table, but it's more than enough to solve the challenge in a few tries.

```
$ make
$ time ./create_rainbow
Building rainbow table with 50000000 chains of length 1000...
Total coverage: 50000000000 hash operations
Rainbow table saved to rainbow_table.txt
./create_rainbow  20009,55s user 35,11s system 1176% cpu 28:24,10 total
```

The final rainbow table look like this :

```
ackees-thunks-879,carpet-insure-676
acinus-lituus-137,tenias-covins-607
access-gained-422,toning-rumina-627
abayas-shlubs-456,precip-cocoon-659
achier-bubkes-76,cnemis-vilify-190
abater-hogged-213,klongs-lentic-356
aboard-naffed-240,halmas-casbah-220
abelia-congou-67,pilots-ductal-857
ackees-thunks-880,kuchen-gibbet-320
abated-achier-174,midget-tetryl-257
acinus-lituus-138,mamees-grapey-131
access-gained-423,cripes-footed-110
abayas-shlubs-457,tanbur-mimbar-884
abodes-veneer-988,mezcal-zippos-263
achier-bubkes-77,meloid-reveal-603
aboard-naffed-241,refelt-richen-466
abelia-congou-68,blowze-wiring-844
...
```

solve.py

Run solve.py. It takes about 1 minute to load the rainbow table into memory. After that, each hash takes 2–3 seconds to check.

```
python3 solve.py
Total operations: 50 000 000 000
Rainbow table found. Loading...
```

```
Online search...

Hash: 71223fb93850fdf27100ddbf1cfdb1ad
No password found after exploring all chains.
Password not found in the rainbow table.

Hash: 01d779628b0fbec4690c32a0e0f00e76
No password found after exploring all chains.
Password not found in the rainbow table.

Hash: ac0ef36ccd05d04a2765c8bd3eeb44bb
No password found after exploring all chains.
Password not found in the rainbow table.

Hash: 7f32fba846197818e8d8b15aab6530e9
No password found after exploring all chains.
Password not found in the rainbow table.

Hash: 76402be50830c4079fc9c0e8e6f6703d
No password found after exploring all chains.
Password not found in the rainbow table.

Hash: f6a290892f484248a98d46ab3cc26694

Password found: chitin-agamid-342
^C
```

You can connect several times by hand with netcat to the challenge and retrieve the hash, then search for it in the rainbow table (using solve.py waiting for you input). This can take from 1 to several tries. If you're unlucky, it may take 10, 20, 30 tries... It all depends on the size of your rainbow table. Here, it was 50 million lines long, with a string of 1000 hashes for each line (i.e. 50 billion possibilities).

You may well succeed by making a rainbow table 2 or 3 times smaller, but you'll need to test more hashes.

It is also possible to automate the process of connecting to the challenge to extract the hash and look it up in the rainbow table automatically in python (without copying the value by hand from one terminal to another, like the solution I propose).

```
$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: 71223fb93850fdf27100ddbf1cfdb1ad

Enter the password: ^C

$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
```

```
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: 01d779628b0fbec4690c32a0e0f00e76

Enter the password: ^C

$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: ac0ef36ccd05d04a2765c8bd3eeb44bb

Enter the password: ^C

$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: 7f32fba846197818e8d8b15aab6530e9

Enter the password: ^C

$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: 76402be50830c4079fc9c0e8e6f6703d

Enter the password: ^C

$ nc amsi-sorbone.fr:4444
=== Welcome to CYBERRUN 1984 ===
You’ve reached the hidden terminal behind the high score machine.
Decrypt the MD5 password to unlock the prototype chip data.
You have 60 seconds. Format: WORD1-WORD2-NUMBER
Example: sauced-betook-220, martin-panino-3, acture-demons-1000

MD5 Hash: f6a290892f484248a98d46ab3cc26694

Enter the password: chitin-agamid-342

✅ Access granted. Unlocking prototype chip...
```

```
FLAG: AMSI{PR...73}
```