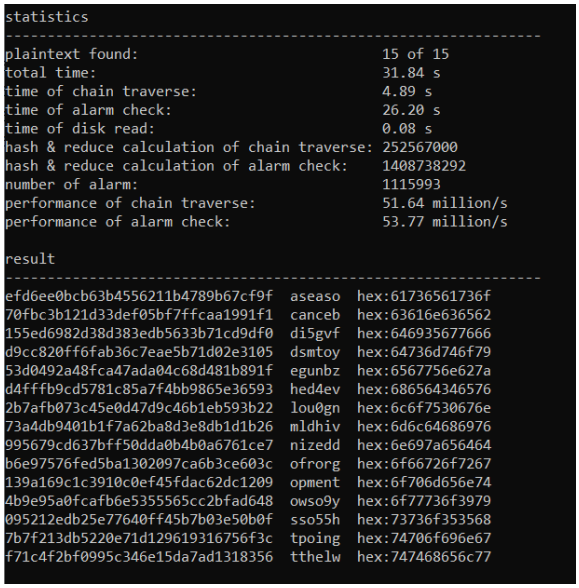
Part 5 – Salting

Sumedha 1002876

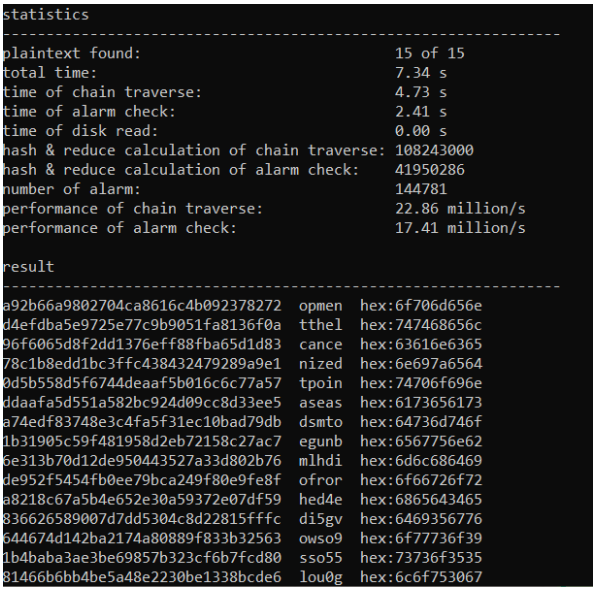
Salted rcrack



```
statistics
---------------------------------------------------------------
plaintext found:                            15 of 15
total time:                                 31.84 s
time of chain traverse:                     4.89 s
time of alarm check:                        26.20 s
time of disk read:                          0.08 s
hash & reduce calculation of chain traverse: 252567000
hash & reduce calculation of alarm check:    1408738292
number of alarm:                            1115993
performance of chain traverse:              51.64 million/s
performance of alarm check:                 53.77 million/s

result
---------------------------------------------------------------
efd6ee0bcb63b4556211b4789b67cf9f  aseaso  hex:61736561736f
70fbc3b121d33def05bf7ffcaa1991f1  canceb  hex:63616e636562
155ed6982d38d383edb5633b71cd9df0  di5gvf  hex:646935677666
d9cc820ff6fab36c7eae5b71d02e3105  dsmtoy  hex:64736d746f79
53d0492a48fca47ada04c68d481b891f  egunbz  hex:6567756e627a
d4fffb9cd5781c85a7f4bb9865e36593  hed4ev  hex:686564346576
2b7afb073c45e0d47d9c46b1eb593b22  lou0gn  hex:6c6f7530676e
73a4db9401b1f7a62ba8d3e8db1d1b26  mldhiv  hex:6d6c64686976
995679cd637bff50dda0b4b0a6761ce7  nizedd  hex:6e697a656464
b6e97576fed5ba1302097ca6b3ce603c  ofrorg  hex:6f66726f7267
139a169c1c3910c0ef45fdac62dc1209  opment  hex:6f706d656e74
4b9e95a0fcafb6e5355565cc2bfad648  owso9y  hex:6f77736f3979
095212edb25e77640ff45b7b03e50b0f  sso55h  hex:73736f353568
7b7f213db5220e71d12961931675 6f3c  tpoing  hex:74706f696e67
f71c4f2bf0995c346e15da7ad1318356  tthelw  hex:747468656c77
```

Non-salted rcrack



```
statistics
---------------------------------------------------------------
plaintext found:                            15 of 15
total time:                                 7.34 s
time of chain traverse:                     4.73 s
time of alarm check:                        2.41 s
time of disk read:                          0.00 s
hash & reduce calculation of chain traverse: 108243000
hash & reduce calculation of alarm check:    41950286
number of alarm:                            144781
performance of chain traverse:              22.86 million/s
performance of alarm check:                 17.41 million/s

result
---------------------------------------------------------------
a92b66a9802704ca8616c4b092378272  opmen  hex:6f706d656e
d4efdba5e9725e77c9b9051fa8136f0a  tthel  hex:747468656c
96f6065d8f2dd1376eff88fba65d1d83  cance  hex:63616e6365
78c1b8edd1bc3ffc438432479289a9e1  nized  hex:6e697a6564
0d5b558d5f6744deaaf5b016c6c77a57  tpoin  hex:74706f696e
ddaafa5d551a582bc924d09cc8d33ee5  aseas  hex:6173656173
a74edf83748e3c4fa5f31ec10bad79db  dsmto  hex:64736d746f
1b31905c59f481958d2eb72158c27ac7  egunb  hex:6567756e62
6e313b70d12de950443527a33d802b76  mlhdi  hex:6d6c686469
de952f5454fb0ee79bca249f80e9fe8f  ofror  hex:6f66726f72
a8218c67a5b4e652e30a59372e07df59  hed4e  hex:6865643465
836626589007d7dd5304c8d22815fffc  di5gv  hex:6469356776
644674d142ba2174a80889f833b32563  owso9  hex:6f77736f39
1b4baba3ae3be69857b323cf6b7fcd80  sso55  hex:73736f3535
81466b6bb4be5a48e2230be1338bcde6  lou0g  hex:6c6f753067
```

Timing:

It takes 31.84s to crack all the salted passwords using the rainbow table which is approximately 4.5 times longer than that of non-salted, 7.34s.

Explanation:

In this lab, Salting adds a random alphabet to the password of the hash function to ensure a unique hash output.

Without salting, hashed passwords are not unique to themselves because of the deterministic and predictable nature of the hash function when given the same input password, thus always giving the same hash output.

An example would be when Alice and Bob choose ‘qwerty’ as their passwords, the hash password generated will be the same for both. As long as the password is found, the attacker is able to access all accounts using the same hash.

By using salt, we are increasing the randomness for each password input. The hash created will be unique for every input even if the input password is the same. With the increased randomness, the hash function will be less predictable and harder to crack.

Simply adding an alphabet to the end of the passwords from the lab, I needed to introduce 4 rainbow tables with table_index (reduction function) 0, 1 and 2 respectively, along with an increased in chain_num from 600 000 to 1 000 000 to find all the salted passwords. Whereas a 0 table_index and chain_num 600 000 is sufficient to crack the non-salted passwords. Salting introduces more complexity when cracking the salted passwords compared to the non-salted passwords as shown from the images above.