

RAINBOWCRACK TUTORIAL

Zhu Shuanglei

RainbowCrack is an instant windows password cracker based on Philippe Oechslin's faster time-memory trade-off technique. It is recommended to look through the paper before you read this tutorial.

In this tutorial, we will guide you through the steps to make things working.

1. Some basis of Time-Memory Trade-Off

There are two typical attacks in cryptanalysis of block ciphers: brute force and table precomputation. In brute force, an attacker tries all possible keys to encrypt a known plaintext for which he has the corresponding ciphertext. The idea of table precomputation is to precompute and store encryptions of a chosen plaintext and corresponding keys for all possible keys.

RainbowCrack use the second method. It precompute and store all possible password - LanManager hash pairs in files so called "rainbow table". Any time the password of a LanManager hash is required, you just query the precomputed tables and find the password in seconds.

2. Select the configuration

First of all, we will select the configuration of the attack. There are so many parameters to be adjusted in the theory: the success rate you want, the character set to use, the hard disk space you can afford and so on. If you know the theory well, you can work on your own. If not, we have prepared some typical parameter configurations for you. They are optimized to the best of my knowledge.

NOTE: All the configurations below are ready for a 666MHz CPU. If your CPU is faster, the performance will be better.

CONFIGURATION #0	
character set	alpha (ABCDEFGHIJKLMNOPQRSTUVWXYZ)
key space	$26^1 + 26^2 + 26^3 + 26^4 + 26^5 + 26^6 + 26^7 = 8353082582$
t	2100
m	8000000
l	5
disk usage	$m * 16 * l = 640000000 \text{ B} = 610 \text{ MB}$
success rate	0.9990
mean cryptanalysis time	3.7841 s
mean cryptanalysis time on a low memory system (free memory size much smaller than 122MB)	8.2836 s
max cryptanalysis time	31.1441 s
table precomputation commands	rtgen alpha 0 2100 8000000 bla rtgen alpha 1 2100 8000000 bla rtgen alpha 2 2100 8000000 bla rtgen alpha 3 2100 8000000 bla rtgen alpha 4 2100 8000000 bla
table precomputation time	2 days 18 hours

RAINBOWCRACK TUTORIAL

Zhu Shuanglei

Some explanations:

character set	we use alpha characters as the plaintext character set
key space	There are 8353082582 different alpha only plaintexts.
t	rainbow chain length, see the paper for detail
m	rainbow chain count of each rainbow table, see the paper for detail
l	rainbow table count, see the paper for detail
disk usage	disk space required to store all generated rainbow tables each rainbow chain will take 16 bytes (8 bytes for a start point and 8 bytes for a end point)
success rate	When the rainbow tables have been generated, you will have the probability 99.9% to crack an alpha only password. Due to the nature of the theory, this is not a granted attack.
mean cryptanalysis time	You need 3.7841 seconds to crack an alpha password on average. It does not take into account the time spent on "false alarm". See the paper to find out what is a "false alarm".
mean cryptanalysis time on a low memory system	If you don't have enough free physical memory to hold one rainbow table a time, the program (rcrack.exe) will have to load the table chunk by chunk and search the table chunk by chunk. Losing the change of finding the password in early time. It does not take into account the time spent on "false alarm".
max cryptanalysis time	If the password you are searching is not covered by the rainbow tables. You will have to search all tables only to find nothing. It does not take into account the time spent on "false alarm".
table precomputation commands	Use the utility "rtgen.exe" in the distribution and these commands to generate the rainbow tables which are required to launch the attack. (see next section of the tutorial for more)
table precomputation time	Table precomputation is time expensive. This is the meaning of "Time-Memory Trade-Off".

CONFIGURATION #1	
character set	alpha-numeric(ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789)
key space	$36^1 + 36^2 + 36^3 + 36^4 + 36^5 + 36^6 + 36^7 = 80603140212$
t	2400
m	40000000
l	5
disk usage	$m * 16 * l = 3200000000 \text{ B} = 3 \text{ GB}$
success rate	0.9904
mean cryptanalysis time	7.6276 s
mean cryptanalysis time on a low memory system (free memory size much smaller than 610MB)	13.3075 s

RAINBOWCRACK TUTORIAL

Zhu Shuanglei

max cryptanalysis time	40.6780 s
table precomputation commands	rtgen alpha-numeric 0 2400 40000000 bla rtgen alpha-numeric 1 2400 40000000 bla rtgen alpha-numeric 2 2400 40000000 bla rtgen alpha-numeric 3 2400 40000000 bla rtgen alpha-numeric 4 2400 40000000 bla
table precomputation time	15 days 17 hours

Some explanations:
With this configuration, you can crack an alpha-numeric password in 13.3075 seconds on a 256MB memory system with 99.04% success rate. Due to the limited CPU power/patience, I accept the 99% success rate instead of the 99.9% one.

In this tutorial we select "CONFIGURATION#0". If you want the second configuration, everything is similar.

3. Precompute the rainbow tables with rtgen.exe

Now the time to generate the rainbow tables.

There is an utility called "rtgen.exe" (rainbow table generator) in the distribution. Now open a MS-DOS prompt, switch to the directory where the rainbowcrack files are extracted, make sure there is 128 MB free disk space in place and execute the command:

```
rtgen alpha 0 2100 8000000 bla
```

This will begin the generation of first rainbow table. It takes 13.2 hours to complete on a 666 MHz CPU.

Leave you computer working ...

You can pause the precomputation any time by pressing Ctrl+C. Next time you run rtgen.exe with the same parameters the program will pick up where the precomputation left off and continue the generation of the rainbow table.

When the generation of first rainbow table is finished. There will be a file named "lm_alpha_0_2100x8000000_bla.rt" (128000000 bytes) in the directory. Don't rename the file because we store some parameters in the file title.

Now the time to generate the remaining rainbow tables, make sure you have enough free disk space (128000000 bytes for each table):

```
rtgen alpha 1 2100 8000000 bla  
rtgen alpha 2 2100 8000000 bla  
rtgen alpha 3 2100 8000000 bla  
rtgen alpha 4 2100 8000000 bla
```

Leave you computer working ...

.....

.....

RAINBOWCRACK TUTORIAL

Zhu Shuanglei

When the precomputation is complete, make sure the following files are in place:

```
128,000,000 bytes  lm_alpha_0_2100x8000000_bla.rt
128,000,000 bytes  lm_alpha_1_2100x8000000_bla.rt
128,000,000 bytes  lm_alpha_2_2100x8000000_bla.rt
128,000,000 bytes  lm_alpha_3_2100x8000000_bla.rt
128,000,000 bytes  lm_alpha_4_2100x8000000_bla.rt
```

If everything goes well, backup all files (recommended) and proceed to the next section of the tutorial.

4. Sort rainbow tables with rtsort.exe

To speed up the search of rainbow table, we should sort the rainbow table with "rtsort.exe" in advance.

In fact "rcrack.exe" only accept sorted rainbow tables.

Use these commands:

```
rtsort lm_alpha_0_2100x8000000_bla.rt
rtsort lm_alpha_1_2100x8000000_bla.rt
rtsort lm_alpha_2_2100x8000000_bla.rt
rtsort lm_alpha_3_2100x8000000_bla.rt
rtsort lm_alpha_4_2100x8000000_bla.rt
```

Each command will take several minutes to complete. The "rtsort.exe" utility will sort the file and write back to the original file.

Notice: If free memory size is smaller than the file size, we can't load the file into memory at a time. In which case extra free disk space as large as the file to be sorted is required to apply an external sort.

If everything goes well, proceed to the next section.

5. Crack the LanManager hash with rcrack.exe and the sorted rainbow tables

Finally you have everything ready. Now the time to play with "rcrack.exe".

Notice the file "random_alpha.txt" in the distribution. It contain hashes of several randomly generated alpha only passwords in pwdump format. We will use this file as the target.

Launch the crack by issuing the command:

```
rcrack c:\rainbowcrack\*.rt -f random_alpha.txt
```

(Replace "c:\rainbowcrack\" with where you place the sorted rainbow tables.)

Have fun!
2003-9-9