

Forum:[Hack Forums](#)

Thread: [Maskprocessor Wordlist Generator](#)

sHyRe - July 4, 2013, 10:43 am

[img]http://www.der-sprachlose.de/bilder/Hacker_DAE2/Hacker.jpg[/img]

What is maskprocessor ? :

Maskprocessor is a High-Performance word generator with a per-position configureable charset packed into a single stand-alone binary.

Where can i download maskprocessor ? :

[Download](#)

How can i use maskprocessor ? :

1. Open up Command Prompt 'CMD'

Start > RUN > cmd.exe

Start > Searchbar > cmd.exe

2. Type in 'cd <FILEPATH>'

Example :

Code cd C:\Users\S0b3r\Downloads\maskprocessor-0.69\maskprocessor-0.69

3. Code Type in 'mp32.exe --help' or 'mp64.exe --help' to open the WIKI/Help File with all useful Commands/Options

4. Decide what you want to build/do with maskprocessor

5. Read the --help File or the instructions in this Tutorial

This Spoiler will help you Spoiler [\(Click to View\)](#)Quote:Usage: mp32/64.exe [options]... mask Startup:

-V, --version Print version

-h, --help Print help

Increment:

-i, --increment Enable increment mode

--increment-min=NUM Start incrementing at NUM

--increment-max=NUM Stop incrementing at NUM

Misc:

--combinations Calculate number of combinations

--hex-charset Assume charset is given in hex

-q, --seq-max Maximum number of multiple sequential characters

Resources:

-s, --start-at=WORD Start at specific position

-l, --stop-at=WORD Stop at specific position

Files:

-o, --output-file=FILE Output-file

Custom charsets:

- 1, --custom-charset1=CS User-defineable charsets
- 2, --custom-charset2=CS Example:
- 3, --custom-charset3=CS --custom-charset1=?dabcdef
- 4, --custom-charset4=CS sets charset ?1 to 0123456789abcdef

* Built-in charsets:

?l = abcdefghijklmnopqrstuvwxyz
?u = ABCDEFGHIJKLMNOPQRSTUVWXYZ
?d = 0123456789
?s = !"#\$%&'()*+,-./:;<=>?@[\\]^_`{|}~
?h = 8 bit characters from 0xc0 - 0xff
?D = 8 bit characters from german alphabet
?F = 8 bit characters from french alphabet
?R = 8 bit characters from russian alphabet

6. Have a look at this modifiable Commands/Options

Spoiler ([Click to View](#))

?l = abcdefghijklmnopqrstuvwxyz
?u = ABCDEFGHIJKLMNOPQRSTUVWXYZ
?d = 0123456789
?s = !"#\$%&'()*+,-./:;<=>?@[\\]^_`{|}~
?h = 8 bit characters from 0xc0 - 0xff
?D = 8 bit characters from german alphabet
?F = 8 bit characters from french alphabet
?R = 8 bit characters from russian alphabet

echo lower

mp32.exe --output-file dicts/bf_l_1.dict ?l
mp32.exe --output-file dicts/bf_l_2.dict ?l?
mp32.exe --output-file dicts/bf_l_3.dict ?l?l?
mp32.exe --output-file dicts/bf_l_4.dict ?l?l?l?

echo digit

mp32.exe --output-file dicts/bf_d_1.dict ?d
mp32.exe --output-file dicts/bf_d_2.dict ?d?d
mp32.exe --output-file dicts/bf_d_3.dict ?d?d?d
mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d

echo upper

mp32.exe --output-file dicts/bf_u_1.dict ?u
mp32.exe --output-file dicts/bf_u_2.dict ?u?u
mp32.exe --output-file dicts/bf_u_3.dict ?u?u?u
mp32.exe --output-file dicts/bf_u_4.dict ?u?u?u?u

echo symbol

```
mp32.exe --output-file dicts/bf_s_1.dict ?s
mp32.exe --output-file dicts/bf_s_2.dict ?s?s
mp32.exe --output-file dicts/bf_s_3.dict ?s?s?s
mp32.exe --output-file dicts/bf_s_4.dict ?s?s?s?s
```

echo lower digit

```
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_1.dict ?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_2.dict ?1?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_4.dict ?1?1?1?1
```

echo lower upper

```
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_1.dict ?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_2.dict ?1?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_4.dict ?1?1?1?1
```

echo lower symbol

```
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_1.dict ?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_2.dict ?1?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_4.dict ?1?1?1?1
```

echo digit upper

```
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_1.dict ?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_2.dict ?1?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_4.dict ?1?1?1?1
```

echo digit symbol

```
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_1.dict ?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_2.dict ?1?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_4.dict ?1?1?1?1
```

echo upper symbol

```
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_1.dict ?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_2.dict ?1?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_3.dict ?1?1?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_4.dict ?1?1?1?1
```

echo lower digit upper

```
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_1.dict ?1
```

```
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_4.dict ?1?1?1?1
```

echo lower digit symbol

```
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_1.dict ?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_4.dict ?1?1?1?1
```

echo lower upper symbol

```
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_1.dict ?1
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_2.dict ?1?1
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_4.dict ?1?1?1?1
```

echo digit upper symbol

```
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_1.dict ?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_2.dict ?1?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_4.dict ?1?1?1?1
```

echo lower digit upper symbol

```
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_1.dict ?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_4.dict ?1?1?1?1
```

echo hex lower

```
mp32.exe --custom-charset1=?dabcdef --output-file dicts/bf_hl_1.dict ?1
mp32.exe --custom-charset1=?dabcdef --output-file dicts/bf_hl_2.dict ?1?1
```

echo hex upper

```
mp32.exe --custom-charset1=?dABCDEF --output-file dicts/bf_hu_1.dict ?1
mp32.exe --custom-charset1=?dABCDEF --output-file dicts/bf_hu_2.dict ?1?1
```

7. Type in 'mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d?d?d?d?d' to create a 8 digit numeric list.

```
--output-file dicts/bf_d_4.dict
--output-file dicts/bf_d_4.dict
```

Red = Name of the output File example --output-file dicts/8digit.lst

How do i modify Commands/Options ? :

This is the Command/Option i pick for showing you how to modify it for you personal needs.

echo digit (for an alphanumeric list)

```
mp32.exe --output-file dicts/bf_d_1.dict ?d (1 newline , 2 newline , 3 newline , 4 newline ...)  
mp32.exe --output-file dicts/bf_d_2.dict ?d?d (01 newline , 02 newline , 03 newline , 04 newline ...)  
mp32.exe --output-file dicts/bf_d_3.dict ?d?d?d (001 newline , 002 newline , 003 newline , 004  
newline ...)  
mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d (0001 newline , 0002 newline , 0003 newline ,  
0004 newline ...)
```

This Commands/Options end with 4 digit '0123' but you need a 8,9 or 10 ... digit list for WPA/WPA2

That means you have to append this

(8 Digit)

Code ?d?d?d?d?d?d?d

(9 Digit)

Code ?d?d?d?d?d?d?d?d

(10 Digit)

Code ?d?d?d?d?d?d?d?d?d

after this Code 'mp32.exe --output-file dicts/bf_d_4.dict'

Example :

Code mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d?d?d?d?d

to create a 8 Digit List (12345678)

Information about the size of this lists :

8 Digit (976.563 KB) < 1 Gigabyte

9 Digit (10.742.188 KB) > 10 Gigabyte

10 Digit (117.187.500 KB) > 117 Gigabyte

I hope i helped some of you have a nice day

PinkPanther - July 4, 2013, 11:57 am

(Jul 4, 2013 10:43 AM)sHyRe Wrote: [\[CITE\]](#) 7. Type in 'mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d?d?d?d' to create a 8 digit alphanumeric list.

That wouldn't be alphanumeric lol. But thanks for this, I was wondering when someone would do a tut on it!

sHyRe - July 4, 2013, 10:20 pm

(Jul 4, 2013 11:57 AM)PinkPanther Wrote: [\[CITE\]](#) That wouldn't be alphanumeric lol. But thanks for this, I was wondering when someone would do a tut on it!

Fixed that thanks for your feedback.

greatwolf - July 6, 2013, 1:22 pm

Can you also go over how to generate other character set permutations besides digits? Also can maskprocessor be used to generate a new wordlist as output given 2 wordlist as input where each line in the new list are 2 words concated together?

I guess another useful thing to add to the tutorial is how to pipe maskprocessor's output directly into a cracking program's input. This way mp can iterate through the given pattern's keyspace on-the-fly and it won't take up any extra storage space.

sHyRe - July 10, 2013, 11:12 am

Have a look at this modifiable Commands/Options
Spoiler (Click to Hide)

```
?l = abcdefghijklmnopqrstuvwxyz
?u = ABCDEFGHIJKLMNOPQRSTUVWXYZ
?d = 0123456789
?s = !"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~
?h = 8 bit characters from 0xc0 - 0xff
?D = 8 bit characters from german alphabet
?F = 8 bit characters from french alphabet
?R = 8 bit characters from russian alphabet
```

echo lower

```
mp32.exe --output-file dicts/bf_l_1.dict ?l
mp32.exe --output-file dicts/bf_l_2.dict ?l?l
mp32.exe --output-file dicts/bf_l_3.dict ?l?l?l
mp32.exe --output-file dicts/bf_l_4.dict ?l?l?l?l
```

echo digit

```
mp32.exe --output-file dicts/bf_d_1.dict ?d
mp32.exe --output-file dicts/bf_d_2.dict ?d?d
mp32.exe --output-file dicts/bf_d_3.dict ?d?d?d
mp32.exe --output-file dicts/bf_d_4.dict ?d?d?d?d
```

echo upper

```
mp32.exe --output-file dicts/bf_u_1.dict ?u
mp32.exe --output-file dicts/bf_u_2.dict ?u?u
mp32.exe --output-file dicts/bf_u_3.dict ?u?u?u
mp32.exe --output-file dicts/bf_u_4.dict ?u?u?u?u
```

echo symbol

```
mp32.exe --output-file dicts/bf_s_1.dict ?s
mp32.exe --output-file dicts/bf_s_2.dict ?s?s
mp32.exe --output-file dicts/bf_s_3.dict ?s?s?s
mp32.exe --output-file dicts/bf_s_4.dict ?s?s?s?s
```

echo lower digit

```
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_1.dict ?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_2.dict ?1?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d --output-file dicts/bf_ld_4.dict ?1?1?1?1
```

echo lower upper

```
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_1.dict ?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_2.dict ?1?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?u --output-file dicts/bf_lu_4.dict ?1?1?1?1
```

echo lower symbol

```
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_1.dict ?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_2.dict ?1?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?s --output-file dicts/bf_ls_4.dict ?1?1?1?1
```

echo digit upper

```
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_1.dict ?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_2.dict ?1?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?u --output-file dicts/bf_du_4.dict ?1?1?1?1
```

echo digit symbol

```
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_1.dict ?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_2.dict ?1?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?s --output-file dicts/bf_ds_4.dict ?1?1?1?1
```

echo upper symbol

```
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_1.dict ?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_2.dict ?1?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_3.dict ?1?1?1
mp32.exe --custom-charset1=?u?s --output-file dicts/bf_us_4.dict ?1?1?1?1
```

echo lower digit upper

```
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_1.dict ?1
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?u --output-file dicts/bf_ldu_4.dict ?1?1?1?1
```

echo lower digit symbol

```
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_1.dict ?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?s --output-file dicts/bf_lds_4.dict ?1?1?1?1
```

echo lower upper symbol

```
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_1.dict ?1
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_2.dict ?1?1
```

```
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?u?s --output-file dicts/bf_lus_4.dict ?1?1?1?1
```

echo digit upper symbol

```
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_1.dict ?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_2.dict ?1?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_3.dict ?1?1?1
mp32.exe --custom-charset1=?d?u?s --output-file dicts/bf_dus_4.dict ?1?1?1?1
```

echo lower digit upper symbol

```
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_1.dict ?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_2.dict ?1?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_3.dict ?1?1?1
mp32.exe --custom-charset1=?l?d?u?s --output-file dicts/bf_ldus_4.dict ?1?1?1?1
```

echo hex lower

```
mp32.exe --custom-charset1=?dabcdef --output-file dicts/bf_hl_1.dict ?1
mp32.exe --custom-charset1=?dabcdef --output-file dicts/bf_hl_2.dict ?1?1
```

echo hex upper

```
mp32.exe --custom-charset1=?dABCDEF --output-file dicts/bf_hu_1.dict ?1
mp32.exe --custom-charset1=?dABCDEF --output-file dicts/bf_hu_2.dict ?1?1
```

If you're using Hashcat as cracking program you dont really need to create lists with maskprocessor because maskprocessor is already built into hashcat and you can try all combinations on the fly by using the mask feature

sHyRe - February 12, 2014, 6:26 am

Noone using this tool in hackforums?