JTR CHEAT SHEET

This cheat sheet presents tips and tricks for using JtR

JtR Community Edition - Linux

Download the JtR Bleeding Jumbo edition with improved capabilities and other goodies.

```
git clone
https://github.com/magnumripper/JohnTheR
ipper -b bleeding-jumbo
```

Compile JtR and enable/disable required features

```
cd JohnTheRipper/
cd src/
./configure
make clean && make -s
```

```
Enable bash completion. add the
following line to your ~/.bashrc
. <JtR path>/run/john.bash_completion
```

Cracking Modes

Wordlist Mode (dictionary attack)

```
./john --wordlist=password.lst hashfile
```

Mangling Rules Mode (hybrid)

```
./john --wordlist=password.lst –
rules:<rulename> hashfile
```

Incremental mode (Brute Force)

```
./john --incremental hashfile
```

External mode (use a program to generate guesses)

```
./john --external: <rulename> hashfile
```

```
Loopback mode (use POT as wordlist)
./john --loopback hashfile
```

```
Mask mode (read MASK under /doc)
./john --mask=?1?1?1?1?1?1?1?1 -1=[A-Z]
hashfile -min-len=8
```

```
Hybrid Mask mode
./john -w=password.lst -
mask='?l?l?w?l?l' hashfile
```

```
Markov mode (Read MARKOV under /doc).
First generate Markov stats:
./calc_stat wordlist markovstats
Then run:
./john -markov:200 -max-len:12 hashfile
--mkv-stats=markovstats
```

```
Prince mode  (Read PRINCE under /doc)
./john --prince=wordlist hashfile
```

Most modes have Maxlen=13 in John.conf but it can be overwritten with `-max-len=N` up to 24

Multiple CPU or GPU

List OpenCL devices and get the device id

```
    ./john --list=opencl-devices
```

List formats supported by OpenCL

```
    ./john --list=formats --
    format=opencl
```

Multiple GPU's

```
    ./john hashes --
    format:<openclformat> --wordlist:<>
    --rules:<> --dev=0,1 --fork=2
```

Multiple CPU's (e.g., 4 cores)

```
    ./john hashes --wordlist:<> --
    rules:<> --dev=2 --fork=4
```

Rules

```
--rules:Single
```

```
--rules:Wordlist
```

```
--rules:Extra
```

`--rules:Jumbo` (all the above)

```
--rules:KoreLogic
```

`--rules:All` (all the above)

Incremental Modes (Brute Force)

```
--incremental:Lower (26 char)
```

```
--incremental:Alpha (52 char)
```

```
--incremental:Digits (10 char)
```

```
--incremental:Alnum (62 char)
```

Incremental mode with new charsets

Create a new charset based on john.pot

```
./john --make-charset=charset.chr
```

Create a new entry in John.conf to accommodate the new charset

```
# Incremental modes
[Incremental:charset]
File = $JOHN/charset.chr
MinLen = 0
MaxLen = 31
CharCount = 95
```

Run JtR with the new charset

```
./john --incremental=charset hashfile
```

Wordlists

Sort a wordlist to use with wordlist rule mode

```
$tr A-Z a-z < SOURCE | sort -u > TARGET
```

Use a POT file to generate a new wordlist

```
cut -d: -f2 john.pot | sort -u > pot.dic
```

Generate candidate passwords for slow hashes.

```
./john --wordlist= password.lst --stdout
--rules:Jumbo | ./unique -mem=25
wordlist.uniq
```

Use external mode for complex rules

http://www.lanmaster53.com/2011/02/creating-complex-password-lists-with-john-the-ripper/

Generate a wordlist that meets the complexity specified in the complex filter

./john --wordlist=[path to word list] --stdout --external:[filter name] > [path to output list]

Try sequences of adjacent keys on a keyboard as candidate passwords

`john` --external:Keyboard `hashfile`

Configuration Items on John.conf

When using both CPU and GPU set this flag

Idle = N

Hidden Options

```
./john --list=hidden-options
```

Display guesses

```
./john --incremental:Alpha -stdout -
session=s1
```

Generate guesses with external program

```
crunch 1 6 abcdefg | ./john hashes -
stdin -session=s1
```

Session and Restore

```
./john hashes -session=name
```

./john --restore:name

Show cracked passwords

```
./john hashes --pot=<>  --show
```

Resources

John-Users Mailing List
http://www.openwall.com/lists/john-users/

Authored by Luis Rocha. This cheat sheet was reviewed by John-Users. It’s distributed according to the Creative Commons v3 “Attribution” License. You’re looking at version 1.0 of this document.

JtR Community Wiki
http://openwall.info/wiki/john

`Documentation under doc folder`

Matt Weir Blog
http://reusablesec.blogspot.ch/

Simple Rule in John.conf

```
[List.Rules:Tryout]
l
u
c
l r
l Az"2015"
d
l A0"2015"
A0"#"Az"#"
```

Details

```
# convert to lowercase
l

# convert to uppercase
u

#capitalize
c

#lowercase the word and reverse it (palindrome)
l r

#lowercase the word and append at end of the word (Az) the number 2015
l Az"2015"

# duplicate
d

# lowercase the word and prepend at beggining of the word (A0) the number 2015
l A0"2015"

Add # to the beginning and end of the word
A0"#"Az"#"
```

Use the Wordlist Rule

Display the password candidates generated with the mangling rule

```
./john --wordlist=password.lst --stdout --rules:Tryout
```

Generate password candidates max length of 8

```
./john --wordlist=password.lst --stdout=8 --rules:Tryout
```

```
./john hashes --wordlist=password.lst --rules:Tryout
```

Simple Wordlist Rules

```
#lowercase the first character, and uppercase the rest
C

#toggle case of all characters in the word
t

#toggle case of the character in position N
TN

#reverse: "Fred" -> "derF"
r

#duplicate: "Fred" -> "FredFred"
d

#reflect: "Fred" -> "FredderF"
f

#rotate the word left: "jsmith" -> "smithj"
{

#rotate the word right: "smithj" -> "jsmith"
}

#append character X to the word
$X

#prefix the word with character X
^X
```

Insert and Delete Wordlist Rules

```
#Remove the first char from the word
[

#Remove the last char from the word
]

#delete the character in position N
DN

#extract substring from position N for up to M characters
xNM

#insert character X in position N and shift the rest right
iNX

#overstrike character in position N with character X
oNX
```

Charset and Conversion Wordlist Rules

```
#shift case: "Crack96" -> "cRACK(^"
S

#lowercase vowels, uppercase consonants: "Crack96" -> "CRaCK96"
V

#shift each character right, by keyboard: "Crack96" -> "Vtsvl07"
R

#shift each character left, by keyboard: "Crack96" -> "Xeaxj85"
L
```

Length control

```
#reject the word unless it is less than N characters long
<N

#reject the word unless it is greater than N characters long
>N

#truncate the word at length N
'N
```

Dictionaries

Generate wordlists from Wikipedia pages: `wget https://raw.githubusercontent.com/zombiesam/wikigen/master/wwg.py`

```
python wwg.py -u http://pt.wikipedia.org/wiki/Fernando_Pessoa -t 5 -o fernandopessoa -m3
```

Generate wordlists from Aspell Dict’s

```
aspell dump dicts
```

```
sudo apt-get install aspell-es
```

```
aspell -d es dump master | aspell -l es expand | awk 1 RS=" |\n" > Spanish.dic
```

Resources

Full Rules Documentation
`http://www.openwall.com/john/doc/RULES.shtml`

Password Analysis and Cracking Kit
`https://thesprawl.org/projects/pack/`

Mangling Rules Generation by Simon Marechal
`http://www.openwall.com/presentations/Passwords12-Mangling-Rules-Generation/`