John the Ripper

- Kurzakte -

Anwendung mit den Default-Optionen von john

- john wählt automatisch den richtigen Verschlüsselungsalgorithmus für die Hashs
- Drücken einer beliebigen Taste für die Statistikanzeige während der Laufzeit
- Unterbrechen mit Strg + C, wobei das Zwischenergebnis in die Datei ~/.john/john.rec geschrieben wird
- zweimaliges Betätigen der Kombination Strg + C bricht john ohne Zwischenspeicherung ab
- gecrackte Passwörter werden in der Standardausgabe angezeigt und in der Datei ~/.john/john.pot gespeichert
- Aufruf mit dem Pfad der Passwortdatei (Kopie) als Option

Benchmark-Test (alle aufgelisteten Formate kann john knacken)

```
linux:~ # john -test
Created directory: /root/.john
Benchmarking: Traditional DES [24/32 4K]... DONE
Many salts:     313779 c/s real, 313779 c/s virtual
...
Benchmarking: NT LM DES [32/32 BS]... DONE
Raw:    3117785 c/s real, 3111562 c/s virtual        # 10 mal schneller als bei DES!
```

Erstellen einer zusammengeführten Passwortdatei

```
linux:~ # unshadow /etc/passwd /etc/shadow >  <passwort-datei>
```

Aufruf von john mit schonendem Ressourcenumgang

```
linux:~ # nice -n 19 john <passwort-datei>
```

- Zwischenstand durch Drücken einer beliebigen Taste

check der Auslastung mit top:

```
 PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
6701 root      25  19  7168 6080 1572 R 99.5  0.6   2:51.90 john
```

Aufruf der gecrackten Passwörter

```
linux:~ # john -show <passwort-datei>
linux:~ # john -show -users:jdoe <passwort-datei>
```

- geknackte Passwörter stehen in Datei john.pot

Weiterführen einer unterbrochenen Sitzung:

```
linux:~ # john –restore
```

Aufruf mit einer speziellen Wortliste und den Wörtebuchregeln

- das Standard-Wörterbuch befindet sich in der Datei /var/lib/john/password.lst

```
linux:~ # john -wordlist:/usr/dict/words -rules passwd
```

nur Passwörter mit einer guten Shell knacken

```
linux:~ # john -w:words.lst -rules -shells:sh,csh,tcsh,bash
```

john mit dem schnellen single-Modus starten:

```
linux:~ # john –single                # nutzt Login/GECOS-Infos als Passwort-Vorgabe
```

john mit dem Brute-Force-Modus starten:

```
linux:~ # john –incremental:LanMan passwortdatei.lanman
```

Feintuning

erfolgt in der Konfigurationsdatei `/var/lib/john/john.conf`
(`/var/lib/john/john.ini` für experimentale Version)

- Festlegen von Regeln für führende und abschließende Zeichen der Passwörter
- Regeln für Wortwiederholungen: testtest **…**

Mailinfo an Benutzer mit schwachem Passwort

```
linux:~ # ./mailer <passwort-datei>
```

ausführliche Hilfe:

```
linux:~ # ls /usr/share/doc/packages/john/
.   CHANGES  CREDITS   EXTERNAL  INSTALL    MODES  OPTIONS  README-1.6.37
..  CONFIG   EXAMPLES  FAQ       LICENSING  NEWS   README   RULES


John the Ripper's Command Line Options
======================================

You can list any number of password files on John's command line, and also
specify some of the following options (all of them are case sensitive, but
can be abbreviated; you can also use the GNU-style long options syntax):

        -single                         "single crack" mode
Enables the "single crack" mode, using rules from [List.Rules:Single].

        -wordfile:FILE                  wordlist mode, read words from FILE,
        -stdin                          or from stdin
These are used to enable the wordlist mode.

        -rules                          enable rules for wordlist mode
Enables wordlist rules, that are read from [List.Rules:Wordlist].

       -incremental[:MODE]             incremental mode [using section MODE]
Enables the incremental mode, using the specified ~/john.ini definition
(section [Incremental:MODE], or [Incremental:All] by default).

        -external:MODE                  external mode or word filter
Enables an external mode, using external functions defined in ~/john.ini's
[List.External:MODE] section.

        -stdout[:LENGTH]                no cracking, write words to stdout
When used with a cracking mode, except for "single crack", makes John
print the words it generates to stdout instead of cracking. While applying
wordlist rules, the significant password length is assumed to be LENGTH,
or unlimited by default.

        -restore[:FILE]                 restore an interrupted session
Continues an interrupted cracking session, reading point information from
the specified file (~/restore by default).

        -session:FILE                   set session file name to FILE
Allows you to specify another point information file's name to use for
this cracking session. This is useful for running multiple instances of
John in parallel, or just to be able to recover an older session later,
not always continue the latest one.

        -status[:FILE]                  print status of a session [from FILE]
Prints status of an interrupted or running session. To get an up to date
status information of a detached running session, send that copy of John
a SIGHUP before using this option.

        -makechars:FILE                 make a charset, overwriting FILE
Generates a charset file, based on character frequencies from ~/john.pot,
for use with the incremental mode. The entire ~/john.pot will be used for
the charset file unless you specify some password files. You can also use
an external filter() routine with this option.

        -show                           show cracked passwords
Shows the cracked passwords in a convenient form. You should also specify
the password files. You can use this option while another John is cracking,
to see what it did so far.

        -test                           perform a benchmark
Benchmarks all the enabled ciphertext format crackers, and tests them for
correct operation at the same time.

        -users:[-]LOGIN|UID[,..]        load this (these) user(s) only
Allows you to filter a few accounts for cracking, etc. A dash before the
list can be used to invert the check (that is, load all the users that
aren't listed).
```

```
        -groups:[-]GID[,..]             load this (these) group(s) only
Tells John to load users of the specified group(s) only.

        -shells:[-]SHELL[,..]           load this (these) shell(s) only
This option is useful to load accounts with a valid shell only, or not to
load accounts with a bad shell. You can omit the path before a shell name,
so '-shells:csh' will match both '/bin/csh' and '/usr/bin/csh', while
'-shells:/bin/csh' will only match '/bin/csh'.

        -salts:[-]COUNT                 set a passwords per salt limit
This feature sometimes allows to achieve better performance. For example
you can crack only some salts using '-salts:2' faster, and then crack the
rest using '-salts:-2'. Total cracking time will be about the same, but
you will get some passwords cracked earlier.

        -format:NAME                    force ciphertext format NAME
Allows you to override the ciphertext format detection. Currently, valid
format names are DES, BSDI, MD5, BF, AFS, LM. You can use this option when
cracking or with '-test'. Note that John can't crack password files with
different ciphertext formats at the same time.

        -savemem:LEVEL                  enable memory saving, at LEVEL 1..3
You might need this option if you don't have enough memory, or don't want
John to affect other processes too much. Level 1 tells John not to waste
memory on login names, so you won't see them while cracking. Higher levels
have a performance impact: you should probably avoid using them unless John
doesn't work or gets into swap otherwise.

 Additional Utilities
----------------------

There're some utilities in John's run directory:

        unshadow PASSWORD-FILE SHADOW-FILE
Combines the passwd and shadow files (when you already have access to
both) for use with John. You might need this since if you only used your
shadow file, the GECOS information wouldn't be used by the "single crack"
mode, and also you wouldn't be able to use the '-shells' option. You'll
usually want to redirect the output of 'unshadow' to a file.

        unafs DATABASE-FILE CELL-NAME
Gets password hashes out of the binary AFS database, and produces a file
usable by John (again, you should redirect the output yourself).

        unique OUTPUT-FILE
Removes duplicates from a wordlist (read from stdin), without changing
the order. You might want to use this with John's '-stdout' option, if
you got a lot of disk space to trade for the reduced cracking time.

        mailer PASSWORD-FILE
A shell script to send mail to all the users who got weak passwords. You
should edit the message inside before using.
linux-fm24:~ # less /usr/share/doc/packages/john/OPTIONS
linux-fm24:~ # cat /usr/share/doc/packages/john/OPTIONS
```