HOWTO zum Übersetzen von "John the Ripper"

(C) 2008-2013 T.Birnthaler/H.Gottschalk <howtos(at)ostc.de>
OSTC Open Source Training and Consulting GmbH
http://www.ostc.de

$Id: make-john-HOWTO.txt,v 1.6 2019/11/26 19:37:07 tsbirn Exp $

Dieses Dokument beschreibt die Übersetzung und Installation des Passwort-Knackprogrammes "John the Ripper" aus den Software-Quellen.

INHALTSVERZEICHNIS

1) Übersetzung und Installation
2) Anwendung

---

1) Übersetzung und Installation

Das Programmpaket "john" in der aktuellen Quellcodeversion aus Internet holen (http://www.openwall.com/john) und im eigenen Heimatverzeichnis ablegen:

```
  john-1.6.tar.gz
```

Archiv ansehen (t=toc) und auspacken (x=extract):

```
  tar tzvf john-1.6.tar.gz   # t=toc, z=compressed, v=verbose, f=archivefile
  tar xzvf john-1.6.tar.gz   # x=extract
```

In entstandenes Unterverzeichnis "john-1.6" wechseln und seinen Inhalt ansehen:

```
  cd john-1.6
  ls -F         # -> README@  doc/  run/  src/
```

Datei "README" ansehen, enthält Hinweise, wie "john" aufzurufen ist und dass die Installationsanleitung in "doc/INSTALL" zu finden ist:

```
  more README
  cd doc
  more INSTALL
```

Datei "INSTALL" enthält Hinweise, wie "john" zu übersetzen ist:

```
  cd ../src
  make                     # -> Liefert Liste von Zielplattformen
  make linux-x86-any-elf   # Möglichkeit 1
  make linux-x86-mmx-elf   # Möglichkeit 2 (schneller als "any")
  make generic             # Möglichkeit 3 (unter Linux nicht zu empfehlen)
```

Beim Aufruf von "make" kommt evtl. die Fehlermeldung, dass der C-Compiler "gcc" nicht gefunden wird. Also diesen als Superuser nachinstallieren (z.B. mit YaST oder "apt-get"):

```
  su + yast + Software installieren + Suchen: gcc + ...
  sudo apt-get install gcc
```

Beim erneuten Aufruf von "make" kommt beim Übersetzen von "bench.c" die Fehlermeldung, dass die Konstante "CLK_TCK" nicht deklariert ist:

```
  bench.c: In function 'benchmark_format':
  bench.c:106: error: 'CLK_TCK' undeclared (first use in this function)
  bench.c:106: error: (Each undeclared identifier is reported only once
  bench.c:106: error: for each function it appears in.)
  bench.c: In function 'benchmark_cps':
  bench.c:147: error: 'CLK_TCK' undeclared (first use in this function)
  make[1]: *** [bench.o] Error 1
  make[1]: Leaving directory `/home/tsbirn/prg/john-1.6/src'
  make: *** [generic.h] Error 1
```

Suchen in den Quellen von "john" erbringt, dass "CLK_TCK" darin mehrfach verwendet, aber nicht definiert wird. Also muss die Konstante in den Standard Include-Dateien des C-Compilers definiert werden. Diese werden ab dem Startverzeichnis "/usr/include" rekursiv nach dem Text "CLK_TCK" durchsucht.

```
  grep -R "CLK_TCK" /usr/include
```

Es gibt einige Include-Dateien, die die Konstante definieren:

```
  /usr/include/d/4.1/std/c/time.d:      const clock_t CLK_TCK        = 1000;
  /usr/include/d/4.1/std/c/time.d:    const clock_t CLK_TCK        = 1000;
  /usr/include/d/4.1/std/c/time.d:    extern clock_t CLK_TCK;
  /usr/include/d/4.1/std/c/time.d:      CLK_TCK = cast(clock_t) sysconf(2);
  /usr/include/d/4.1.3/std/c/time.d:    const clock_t CLK_TCK        = 1000;
  /usr/include/d/4.1.3/std/c/time.d:    const clock_t CLK_TCK        = 1000;
  /usr/include/d/4.1.3/std/c/time.d:    extern clock_t CLK_TCK;
  /usr/include/d/4.1.3/std/c/time.d:    CLK_TCK = cast(clock_t) sysconf(2);
  /usr/include/bash/systimes.h:   errors.  All times are in CLK_TCKths of a second.  */
  /usr/include/bash/posixtime.h:#if !defined (HAVE_SYSCONF) || !defined (_SC_CLK_TCK)
  /usr/include/bash/posixtime.h:#  if !defined (CLK_TCK)
  /usr/include/bash/posixtime.h:#      define CLK_TCK   HZ
  /usr/include/bash/posixtime.h:#      define CLK_TCK   60                      /* 60HZ
*/
  /usr/include/bash/posixtime.h:#  endif /* !CLK_TCK */
  /usr/include/bash/posixtime.h:#endif /* !HAVE_SYSCONF && !_SC_CLK_TCK */
  /usr/include/bits/confname.h:    _SC_CLK_TCK,
  /usr/include/bits/confname.h:#define  _SC_CLK_TCK                     _SC_CLK_TCK
  /usr/include/bits/time.h:/* Even though CLOCKS_PER_SEC has such a strange value
CLK_TCK
  /usr/include/bits/time.h:#   define CLK_TCK ((__clock_t) __sysconf (2))       /* 2 is
_SC_CLK_TCK */
  /usr/include/sys/times.h:   All times are in CLK_TCKths of a second.  */
  /usr/include/time.h:#  ifndef CLK_TCK
  /usr/include/time.h:#   define CLK_TCK        CLOCKS_PER_SEC
```

Nachschlagen in "bench.c" erbringt, dass "time.h" includiert wird, d.h. eigentlich müsste alles in Ordnung sein, ist es aber nicht. Der zugewiesene Wert von "CLOCKS_PER_SEC" scheint 1000 zu sein:

```
  grep -R "CLOCKS_PER_SEC" /usr/include
```

Also als (schmutzigen!) Workaround NACH den Include-Dateien (#include <...>) folgendes in "bench.c" einfügen und nochmal mit "make linux-x86-any-elf" übersetzen:

```
  #define  CLK_TCK  1000          /* Konstante definieren */
```

Analog aufgrund der gleichen Fehlermeldung beim Übersetzen die gleiche Zeile in Datei "status.c" einfügen und "john" lässt sich endlich vollständig übersetzen:

```
  make linux-x86-any-elf
```

Danach befindet sich laut Anleitung "doc/INSTALL" das Programm "john" im Verz. "run":

```
  cd ..              # Zurück in Startverzeichnis
  run/john           # Aufruf mit Pfad (sonst nicht gefunden)
  run/john -test     # Geschwindigkeit testen
```

2) Anwendung

Das Programm als Superuser auf die Datei "/etc/shadow" anwenden, um die Passworte darin zu knacken (lastet Rechner sehr stark aus!):

```
  su
  run/john /etc/shadow
  run/john -wordfile:run/password.lst -rules /etc/shadow
```

Abbruch von "john" erfolgt mit Strg-C. Die bisher geknackten Passworte zeigt man an mit:

```
  run/john -show
```