CYB101 Project 3

(🔗 Instructions Page)

👤 Student Name: mahamadou Nimaga
✉️ Student Email: Mahamadou.nimaga35@gmail.com

Reflection (Required)

> 🤔 **Reflection Question #1:** If I had to **explain "password security" in 3 emojis,** they would be... (Feel free to put other comments about your experience this unit here, too!)

😀🔒🔑 it was great seeing strong passwords.

> 🛡️ **Reflection Question #2:** What factors make a password hash difficult to crack? Does that match with usual "password requirements" for signing up to websites?

A strong password hash is hard to crack when the password is complex, lengthy, and salted, using robust hashing algorithms. This corresponds to typical password requirements on websites, which encourage complex and lengthy passwords to enhance security. Secure password storage practices, like salting and strong hashing, add an extra layer of protection.

> 📣 **Shoutouts:** Share appreciation for anyone who helped you out with this project or made your day a little better!

Password Cracking Challenge (Required)

Use the answer boxes below to input your screenshots and example commands for this project. Clarifying notes are optional.

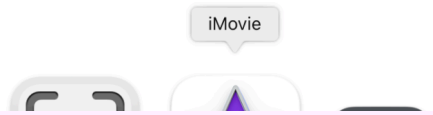
> **Total Number of Cracked Hashes** (Screenshot of output from `john --show CPLeak.txt`)

[Insert Screenshot Here]

```
elsie            (asinghania)
izzy             (nmohd)
loki             (nbowman)
reed             (foliveira)
a                (astringer)
1952             (dhuber)
eieio            (kgomes)
Frank            (vvictor)
kim1             (mcompton)
jazz1            (zshahid)
Bear1            (dpeter)
Anna1            (ppatil)
nairb            (fzahra)
Ailuj            (lray)
paveL            (adavidson)
Susan?           (erosas)
Deans            (bsprague)
Dans             (sfathy)
treed            (dvelez)
Proceeding with incremental:ASCII
manda            (asullivan)
ashia            (mashton)
manch            (amcclain)
metra            (croper)
amaya            (seast)
micol            (rabdullah)
matto            (tsamson)
stach            (gcassidy)
steen            (yhussain)
sears            (nirwin)
andar            (brao)
sammo            (kdickinson)
jamez            (mmonte)
apart            (avandelay)
suman            (scyr)
shola            (adelarosa)
benda            (jcoates)
babby            (jbuchanan)
braze            (wkhan)
brush            (crhodes)
boote            (rbriones)
breal            (rphillips)
bonds            (blindsay)
camil            (rdonovan)
camon            (kpriya)
cooly            (czuniga)
1181             (mgaffney)
1200             (dbuchanan)
1080             (pvictor)
2343             (jwalden)
alfio            (khackett)
```

Screenshot

```
Proceeding with incremental:ASCII
manda            (asullivan)
ashia            (mashton)
manch            (amcclain)
metra            (croper)
amaya            (seast)
micol            (rabdullah)
matto            (tsamson)
stach            (gcassidy)
steen            (yhussain)
sears            (nirwin)
andar            (brao)
sammo            (kdickinson)
jamez            (mmonte)
apart            (avandelay)
suman            (scyr)
shola            (adelarosa)
benda            (jcoates)
babby            (jbuchanan)
braze            (wkhan)
brush            (crhodes)
boote            (rbriones)
breal            (rphillips)
bonds            (blindsay)
camil            (rdonovan)
camon            (kpriya)
cooly            (czuniga)
1181             (mgaffney)
1200             (dbuchanan)
1080             (pvictor)
2343             (jwalden)
alfio            (khackett)
lillo            (dhull)
lilou            (kkirk)
lisha            (csloan)
lucca            (jsoper)
lunas            (olawal)
1985             (mlodge)
1127             (rkaka)
luli             (dcrowley)
2232             (aloh)
2405             (ssohail)
sings            (lchampion)
singe            (mwhite)
celeb            (akavanagh)
ang              (jsinclair)
sonya            (mrivas)
beto             (avillar)
pung             (mjohnstone)
maiga            (mlarry)
myass            (iperera)
```

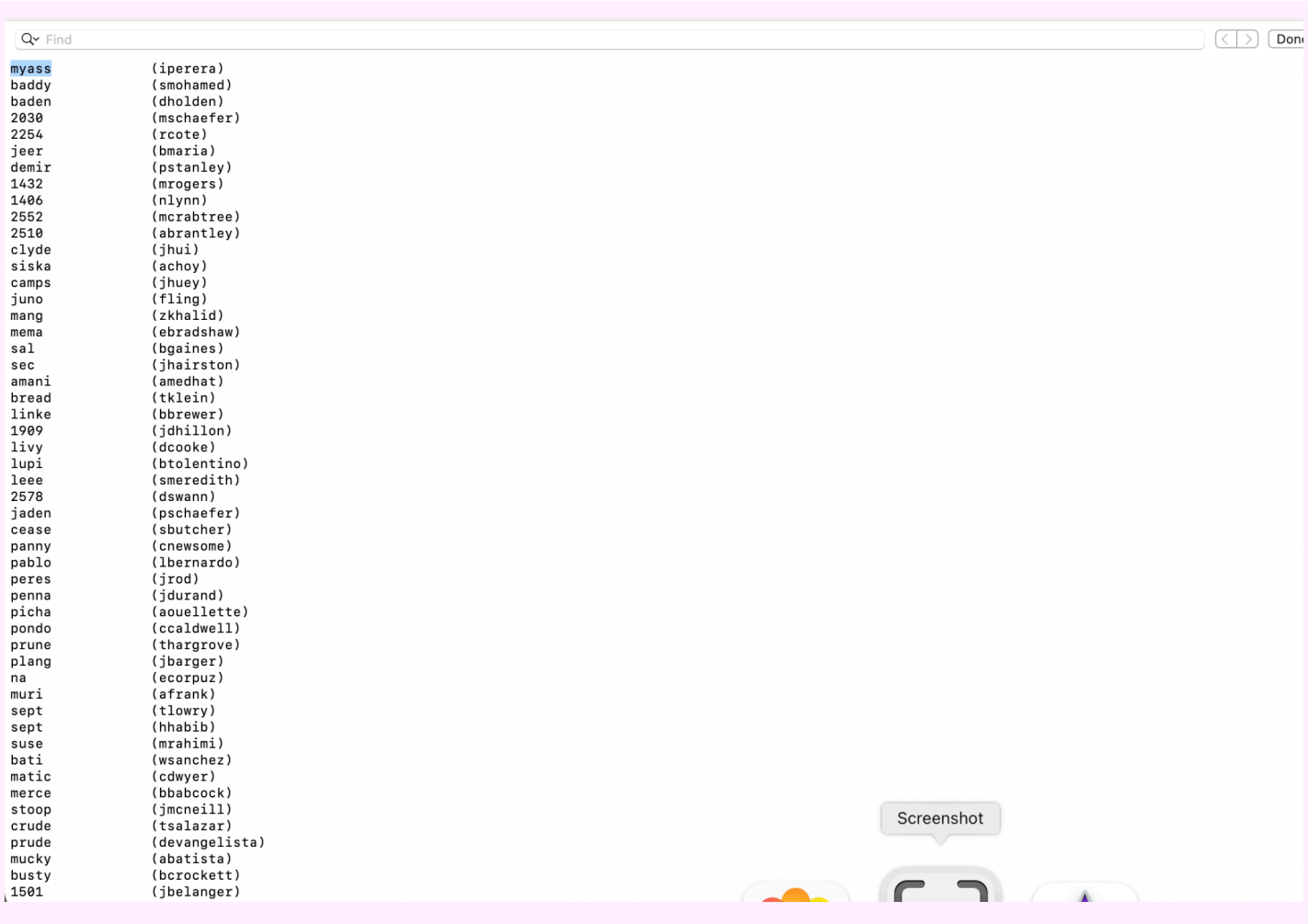


```
zsh completions have been installed to:
  /usr/local/share/zsh/site-functions
==> Summary
🍺  /usr/local/Cellar/john-jumbo/1.9.0_1: 446 files, 75.9MB
==> Running `brew cleanup john-jumbo`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
mahamadou.nimaga@Momos-MBP ~ % cd ~/Downloads

mahamadou.nimaga@Momos-MBP Downloads % pwd

/Users/mahamadou.nimaga/Downloads
mahamadou.nimaga@Momos-MBP Downloads % john CPLeak.txt

Created directory: /Users/mahamadou.nimaga/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-opencl"
Use the "--format=md5crypt-opencl" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1000 password hashes with 1000 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE4.1 4x5])
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 3 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 2 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 20 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 20 needed for performance.
Further messages of this type will be suppressed.
To see less of these warnings, enable 'RelaxKPCWarningCheck' in john.conf
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/local/Cellar/john-jumbo/1.9.0_1/share/john/password.lst, rules:Wordlist
hello            (kpatton)
missy            (yfang)
alice            (dweir)
kitty            (eschroeder)
bingo            (igarza)
lola             (lpadilla)
nick             (cvelazquez)
pluto            (jburnette)
stone            (dkohler)
ace              (htrang)
beer             (jrandolph)
agent            (epetersen)
cleo             (smanalo)
elsie            (asinghania)
```

Screenshot

Notes (Optional)**:**

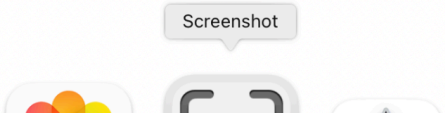

Find
Don
1501 (jbelanger)
1512 (apaige)
siro (dsilverman)
kaina (rshannon)
kemal (mgibbs)
kokan (apangan)
korie (nabraham)
kurtz (daquino)
klara (cworkman)
mex (gmartinez)
1914 (lhaynes)
tic (tschmitz)
busa (jchoy)
2376 (brubio)
soft (swebster)
kelek (ralves)
kokot (mmcclendon)
bravo (ttripp)
crime (ngordon)
danon (jtyree)
dulse (sgregg)
teeny (skhawaja)
tasty (lstuart)
tatti (amackenzie)
tatum (jaltman)
tamal (pdias)
taryn (jblair)
table (lbray)
tazer (rmendiola)
tinka (aspringer)
tinge (trobertson)
torey (sdowning)
trode (smarrero)
tulum (sdunn)
tutee (aszabo)
twiny (rreece)
1828 (kbraun)
plow (lparr)
2704 (nnixon)
slap (spena)
u (bcordova)
dim (mgreenberg)
tro (sjoshi)
meiko (pkannan)
saura (vmarshall)
dobro (thenderson)
paddi (jmarriott)
pisay (dwilder)
anay (rcunanan)
adia (jzeng)
1907 (anewby)
Screenshot

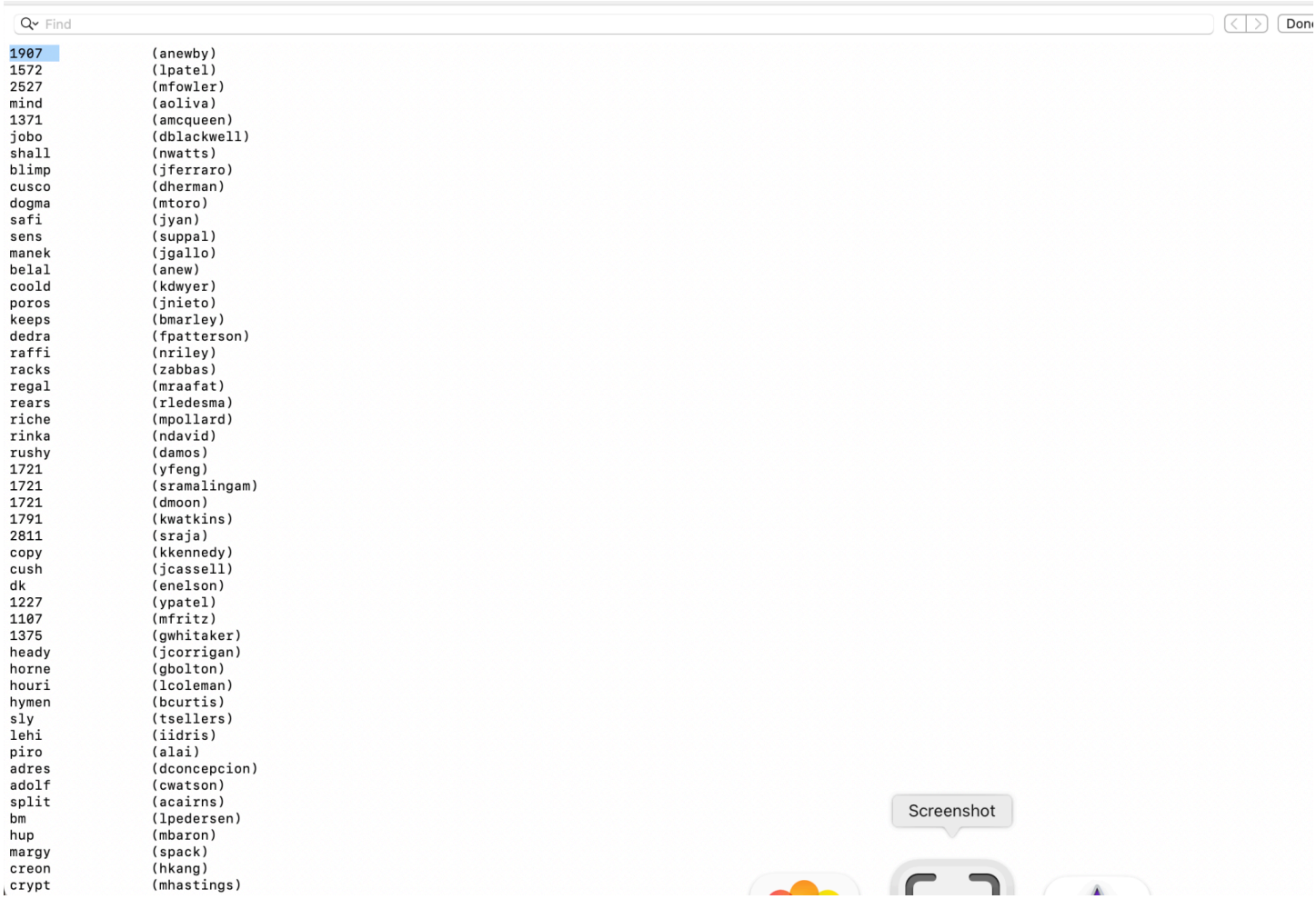

Find
Don
myass (iperera)
baddy (smohamed)
baden (dholden)
2030 (mschaefer)
2254 (rcote)
jeer (bmaria)
demir (pstanley)
1432 (mrogers)
1406 (nlynn)
2552 (mcrabtree)
2510 (abrantley)
clyde (jhui)
siska (achoy)
camps (jhuey)
juno (fling)
mang (zkhalid)
mema (ebradshaw)
sal (bgaines)
sec (jhairston)
amani (amedhat)
bread (tklein)
linke (bbrewer)
1909 (jdhillon)
livy (dcooke)
lupi (btolentino)
leee (smeredith)
2578 (dswann)
jaden (pschaefer)
cease (sbutcher)
panny (cnewsome)
pablo (lbernardo)
peres (jrod)
penna (jdurand)
picha (aouellette)
pondo (ccaldwell)
prune (thargrove)
plang (jbarger)
na (ecorpuz)
muri (afrank)
sept (tlowry)
sept (hhabib)
suse (mrahimi)
bati (wsanchez)
matic (cdwyer)
merce (bbabcock)
stoop (jmcneill)
crude (tsalazar)
prude (devangelista)
mucky (abatista)
busty (bcrockett)
1501 (jbelanger)
Screenshot

Find

Done

```
creon           (hkang)
crypt           (mhastings)
claus           (sregmi)
louka           (efleming)
penta           (srex)
rik             (jbreeden)
1616            (smuir)
ardis           (kramos)
hires           (aswann)
291815          (hevans)
toop            (hobrien)
tink            (hcameron)
tice            (fholmes)
tron            (wevans)
tube            (mpepper)
blaa            (kmcknight)
jun             (bmcmillan)
rmc             (sburnett)
dubhe           (sledbetter)
ayaz            (pchand)
ayla            (rtomas)
kaki            (jryan)
kron            (idavid)
klar            (athomson)
skell           (lvance)
dwain           (lbingham)
psalm           (tzhao)
aki             (sredding)
coo             (nlie)
cce             (kwilkes)
rubik           (jnolasco)
nadie           (lestes)
nacho           (sjayne)
nalin           (pbailey)
namer           (sfigueroa)
nukes           (demery)
dore            (nsuresh)
dong            (hrichards)
dink            (jculver)
ad              (bw)
skey            (dhaskins)
gatta           (dosorio)
garam           (dfinn)
gammy           (sbean)
grill           (iraza)
goony           (dewing)
gohan           (afahad)
geras           (tferrer)
genia           (lbelanger)
geste           (aforde)
```





Sample `john` commands

Please type or copy-and-paste your commands into the boxes below. You do not need to include screenshots of running the commands.

> **Command #1:** a `john` command using **a different wordlist** (not lower.lst)

```
john --wordlist=/path/to/your/wordlist.txt CPLeak.txt
```

> **Command #2:** a `john` command using **a built-in ruleset**

```
john --rules=best64 CPLeak.txt
```

> **Command #3:** a `john` command using **a custom mask**

> **john --mask=?d?d?d?d?d CPLeak.txt**

Submission Checklist

👉*Check off each of the features you have completed.* ***You will only be graded on the features you check off.***

Reflection

- [x] ~~Reflection Question #1 answered above~~
- [x] ~~Reflection Question #2 answered above~~

Required Challenge

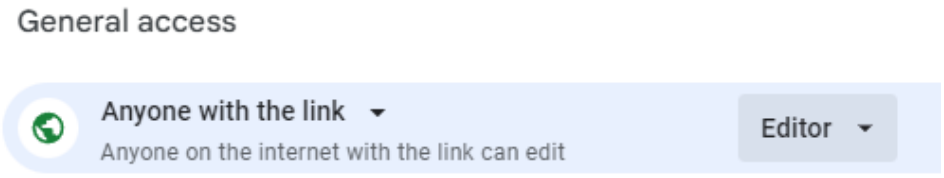
- [x] ~~Screenshot of Total Passwords Cracked~~
    - [ ] 250+ Passwords Cracked
- [x] ~~Command #1~~
- [x] ~~Command #2~~
- [x] ~~Command #3~~

Stretch Challenge

- [ ] 500+ Passwords Cracked
- [ ] 750+ Passwords Cracked
- [ ] 1000 Passwords Cracked

Submit your work!

Step 1: **Click** the Share button at the top of your screen double check that anyone with the link can edit. (This allows our grading team to input your grade below!)



Step 2: **Copy** the link to this document.

Step 3: **Submit** the link on the portal.

Grader Comments

Once your project has been assessed, our graders will leave feedback for you in this space. Please do not delete.

Grading Rubric

Reflection Questions	Total Received	Total Possible
Reflection Question #1 answered above	2	2
Reflection Question #2 answered above	2	2
PART A TOTAL	4	**4**
Password Cracking Challenge	Total Received	Total Possible
Screenshot of Total Passwords Cracked, showing 250+ Cracks	5	7
Command #1	3	3
Command #2	3	3
Command #3	3	3

PART B TOTAL	14	**16**
Stretch Challenge	Total Received	Total Possible
500+ Passwords Cracked	0	+1 bonus
750+ Passwords Cracked	0	+1 bonus
1000 Passwords Cracked	0	+1 bonus
Total Possible Points (Part A + Part B)	18	**20** (+3)

Grader Feedback