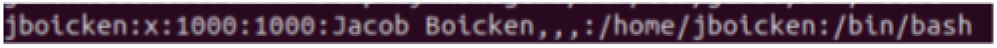
Screenshot of `passwd` and `shadow` with description/notation of fields

Password indicator

UID and GID values for the user and their default group

Home Directory

Default login shell



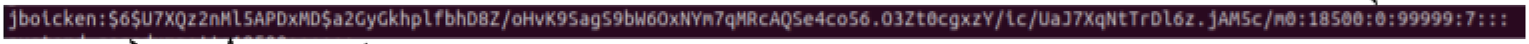
```
jboicken:x:1000:1000:Jacob Boicken,,,:/home/jboicken:/bin/bash
```

Username

Extra comment about the user, i.e. their full name, phone number ...

Minimum days until password can be changed

Maximum days password is valid



```
jboicken:$6$U7XQz2nMl5APDxMD$a2GyGkhplfbhD8Z/oHvK9SagS9bW6OxNYm7qMRcAQSe4co56.O3Zt0cgxzY/lc/UaJ7XqNtTrDl6z.jAM5c/m0:18500:0:99999:7:::
```

Password: $id$salt$hash
My id is 6 so it used sha-512.

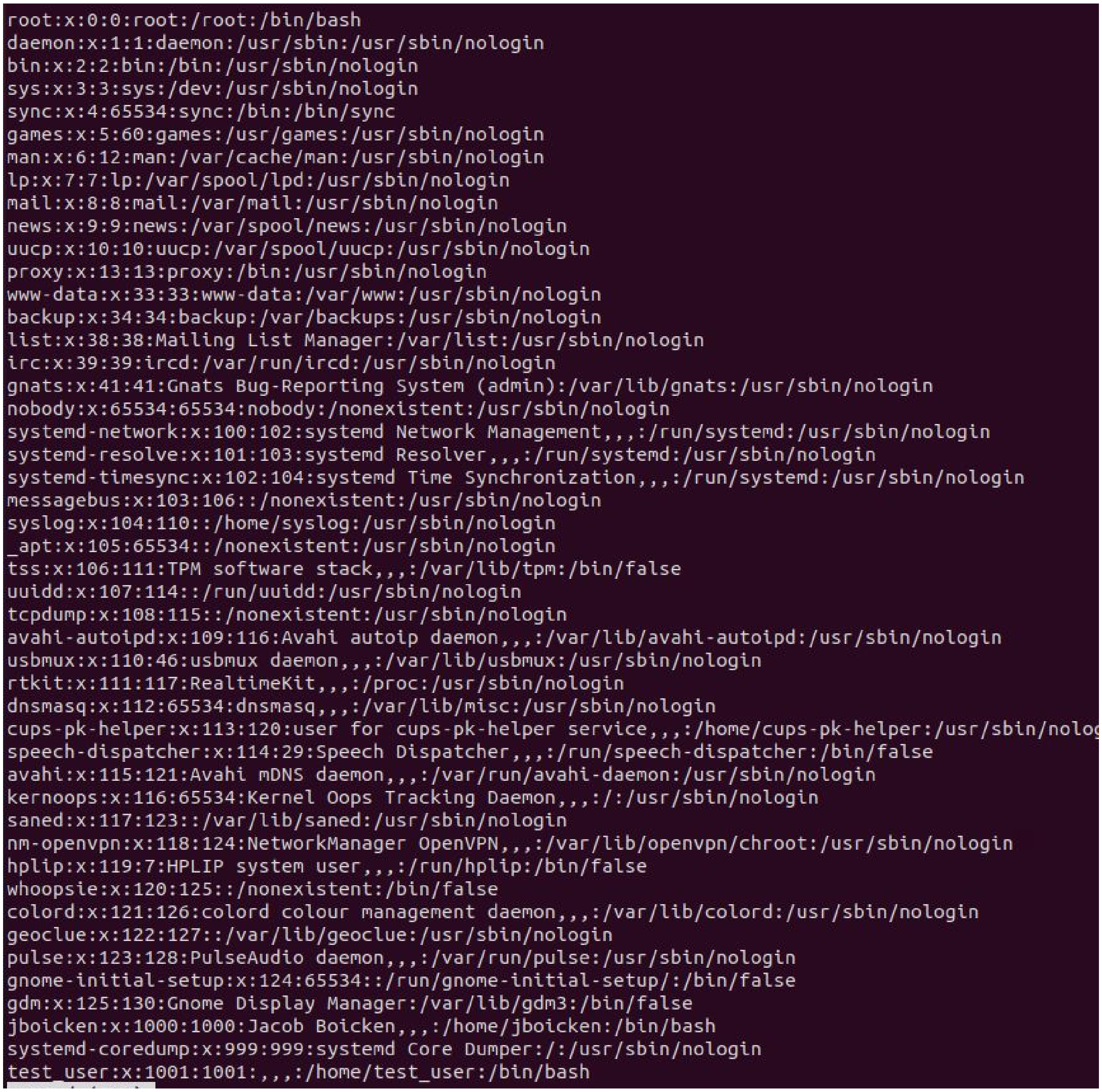
Days since epoch: set when password was last changed

Number of days before to warn of password expiration

The two possible values not in my user's shadow config is the number of days after password expiration that the account is disabled and the number of days after the epoch of when the account is disabled.

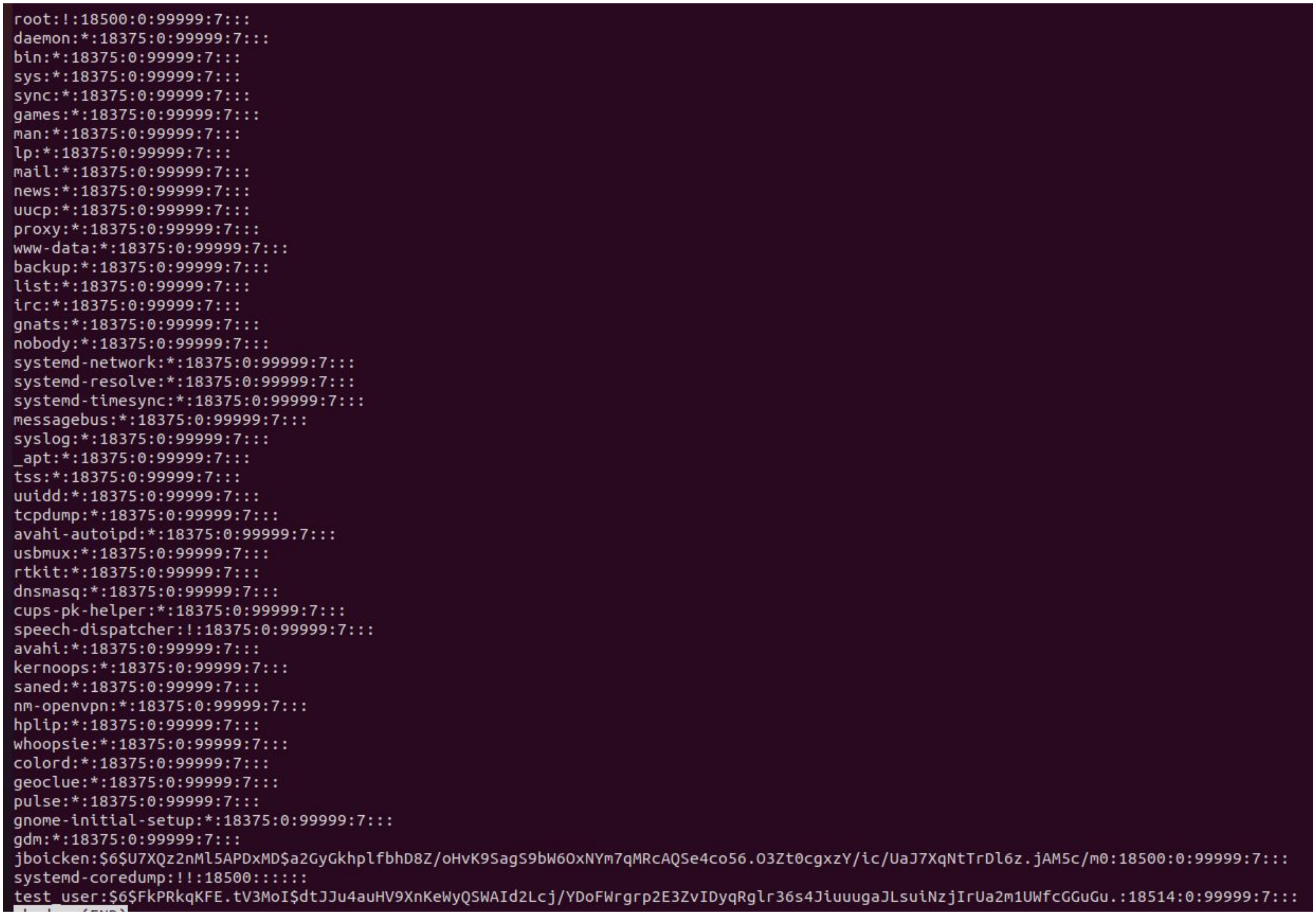
Screenshot of `passwd` and shadow in middle of user account creation

/etc/passwd



```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:110:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:117:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:113:120:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nolog
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
avahi:x:115:121:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/:/usr/sbin/nologin
saned:x:117:123::/var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
hplip:x:119:7:HPLIP system user,,,:/run/hplip:/bin/false
whoopsie:x:120:125::/nonexistent:/bin/false
colord:x:121:126:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:128:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup/:/bin/false
gdm:x:125:130:Gnome Display Manager:/var/lib/gdm3:/bin/false
jboicken:x:1000:1000:Jacob Boicken,,,:/home/jboicken:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
test_user:x:1001:1001:,,,:/home/test_user:/bin/bash
```

/etc/shadow

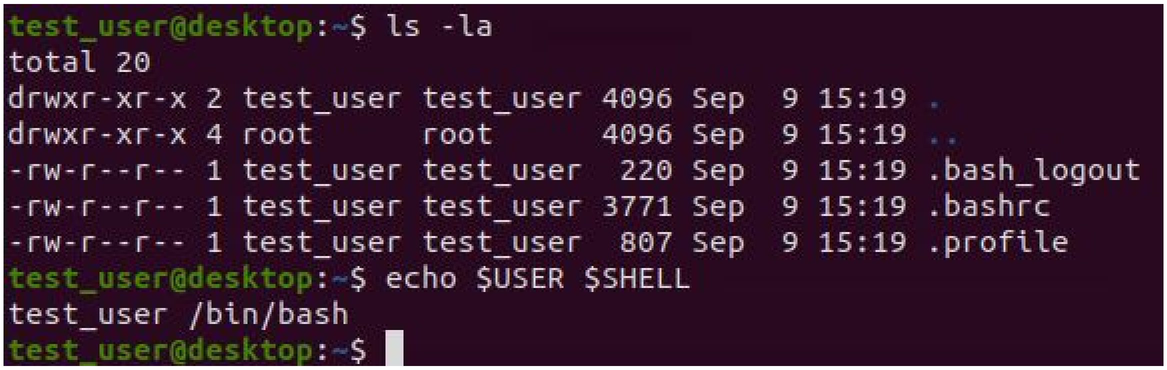


```
root:!:18500:0:99999:7:::
daemon:*:18375:0:99999:7:::
bin:*:18375:0:99999:7:::
sys:*:18375:0:99999:7:::
sync:*:18375:0:99999:7:::
games:*:18375:0:99999:7:::
man:*:18375:0:99999:7:::
lp:*:18375:0:99999:7:::
mail:*:18375:0:99999:7:::
news:*:18375:0:99999:7:::
uucp:*:18375:0:99999:7:::
proxy:*:18375:0:99999:7:::
www-data:*:18375:0:99999:7:::
backup:*:18375:0:99999:7:::
list:*:18375:0:99999:7:::
irc:*:18375:0:99999:7:::
gnats:*:18375:0:99999:7:::
nobody:*:18375:0:99999:7:::
systemd-network:*:18375:0:99999:7:::
systemd-resolve:*:18375:0:99999:7:::
systemd-timesync:*:18375:0:99999:7:::
messagebus:*:18375:0:99999:7:::
syslog:*:18375:0:99999:7:::
_apt:*:18375:0:99999:7:::
tss:*:18375:0:99999:7:::
uuidd:*:18375:0:99999:7:::
tcpdump:*:18375:0:99999:7:::
avahi-autoipd:*:18375:0:99999:7:::
usbmux:*:18375:0:99999:7:::
rtkit:*:18375:0:99999:7:::
dnsmasq:*:18375:0:99999:7:::
cups-pk-helper:*:18375:0:99999:7:::
speech-dispatcher:!:18375:0:99999:7:::
avahi:*:18375:0:99999:7:::
kernoops:*:18375:0:99999:7:::
saned:*:18375:0:99999:7:::
nm-openvpn:*:18375:0:99999:7:::
hplip:*:18375:0:99999:7:::
whoopsie:*:18375:0:99999:7:::
colord:*:18375:0:99999:7:::
geoclue:*:18375:0:99999:7:::
pulse:*:18375:0:99999:7:::
gnome-initial-setup:*:18375:0:99999:7:::
gdm:*:18375:0:99999:7:::
jboicken:$6$U7XQz2nMl5APDxMD$a2GyGkhplfbhD8Z/oHvK9SagS9bW6OxNYm7qMRcAQSe4co56.O3Zt0cgxzY/ic/UaJ7XqNtTrDl6z.jAM5c/m0:18500:0:99999:7:::
systemd-coredump:!!:18500::::::
test_user:$6$FkPRkqKFE.tV3MoI$dtJJu4auHV9XnKeWyQSWAId2Lcj/YDoFWrgrp2E3ZvIDyqRglr36s4JiuuugaJLsuiNzjIrUa2m1UWfcGGuGu.:18514:0:99999:7:::
```

Summarize the key differences you found between `nologin` and a locked password.
(I had this done so it is staying, but shouldn't be graded)

The nologin tells you when the password is inputted correctly that the account is not available to be logged into. However, when the password of a user is locked, then any inputted password is always said to not be correct. Essentially, nologin is a soft lock that prevents the account from being used but the password is inputtable, whereas locking is a hard lock on the account preventing the password to be utilized. (Locking wouldn't let a brute force be possible.)

Screenshot of new user's home directory contents (`ls -la`)



```
test_user@desktop:~$ ls -la
total 20
drwxr-xr-x 2 test_user test_user 4096 Sep  9 15:19 .
drwxr-xr-x 4 root      root      4096 Sep  9 15:19 ..
-rw-r--r-- 1 test_user test_user  220 Sep  9 15:19 .bash_logout
-rw-r--r-- 1 test_user test_user 3771 Sep  9 15:19 .bashrc
-rw-r--r-- 1 test_user test_user  807 Sep  9 15:19 .profile
test_user@desktop:~$ echo $USER $SHELL
test_user /bin/bash
test_user@desktop:~$
```

Description of the effect of each of the (5) `chmod` commands

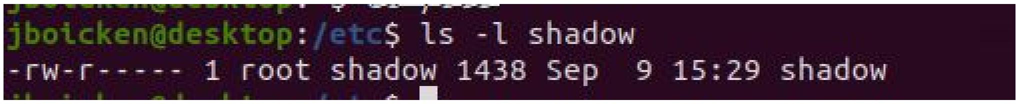
Command	Effect
`chmod 777` *`filename`*	File owner (UID), group (GID), and others are set to read, write and execution privileges
`chmod 700` *`filename`*	Only the file owner is set to be able to read, write, and execute the file. Everyone else is set to no access in any way.
`chmod u=rw` *`filename`*	Sets the file owners permissions to read and write only. Everyone else is unchanged.
`chmod go+x` *`filename`*	Adds execution privileges to the group and others permissions. The file owner is unchanged.
`chmod a+w` *`filename`*	Adds write privileges to everyone (the user, group, and others).

<u>Organized</u> description of directory permissions and how affects contents

Privilege of Directory	Effects
Read	Allows reading what contents are in the directory.
Write	Allows creating, deleting, and editing files & subdirectories under the directory
Execute	Allows opening the directory, which is needed in order to read & write. This allows reading of the files & subdirectories within a directory, if that file or subdirectory allows the user to read it.

Screenshot and description of `/etc/shadow` file - who can r/w/x and why it's set this way.

/etc/shadow only allows for the root user to edit and read the file and a shadow group to read the file. I think this is probably used for authentication to not have to happen via root. That way some utilities like authentication can be run as part of the shadow group to find what hash function to use and then compare your hashed password to a new generated hash from input.

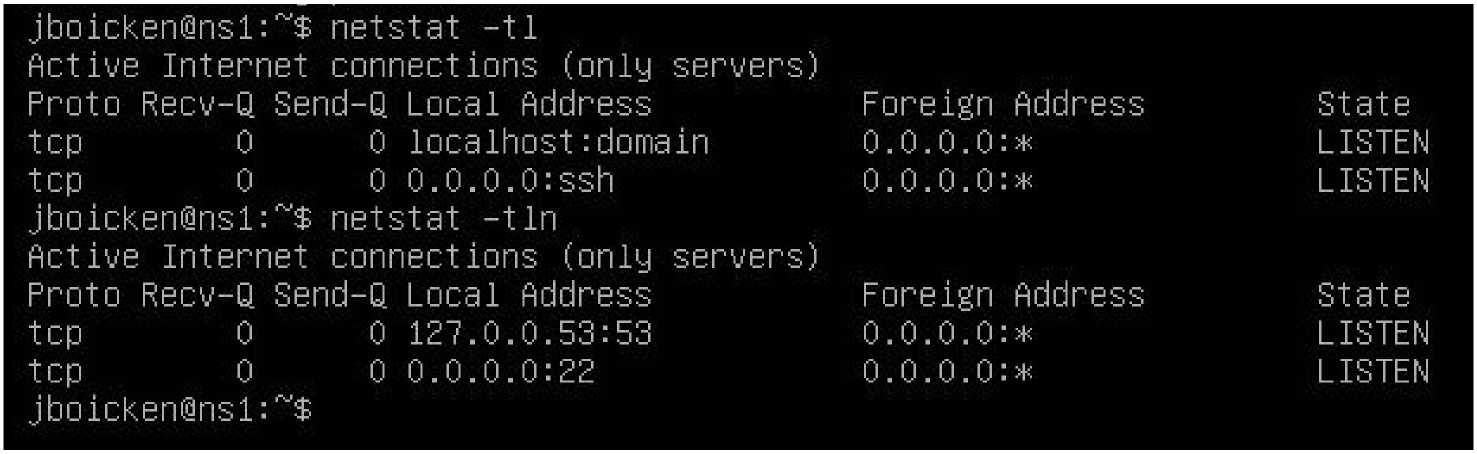


```
jboicken@desktop:/etc$ ls -l shadow
-rw-r----- 1 root shadow 1438 Sep  9 15:29 shadow
```

Meanings of `dig` queries
(The -t specifies the type of the query.)

Query	What does the query result mean?
`dig -t mx iastate.edu`	It checks for mail transfer agents under the iastate.edu domain.
`dig -t ns iastate.edu`	It checks for the authoritative name servers in the dns zone of iastate.edu. I believe authoritative name servers means the ones that will return definitive answers to a DNS query.
`dig -t soa iastate.edu`	It returns information on the dns zone of iastate.edu. It shows the main name server, the domain's admin, and a timestamp that changes upon update. As well, timers that mean the following: ● The number of seconds before the zone should be refreshed. ● The seconds until a failed refresh should be retried. ● The max seconds before a zone stops being authoritative. ● A negative result's Time To Live, TTL. How long a negative DNS query can be cached for Src: https://support.dnsimple.com/articles/soa-record/
`dig -t aaaa iastate.edu`	It gets the ipv6 address of iastate.edu.
`dig -t any iastate.edu`	It gets all record types about iastate.edu

Screenshot of the (2) netstat outputs

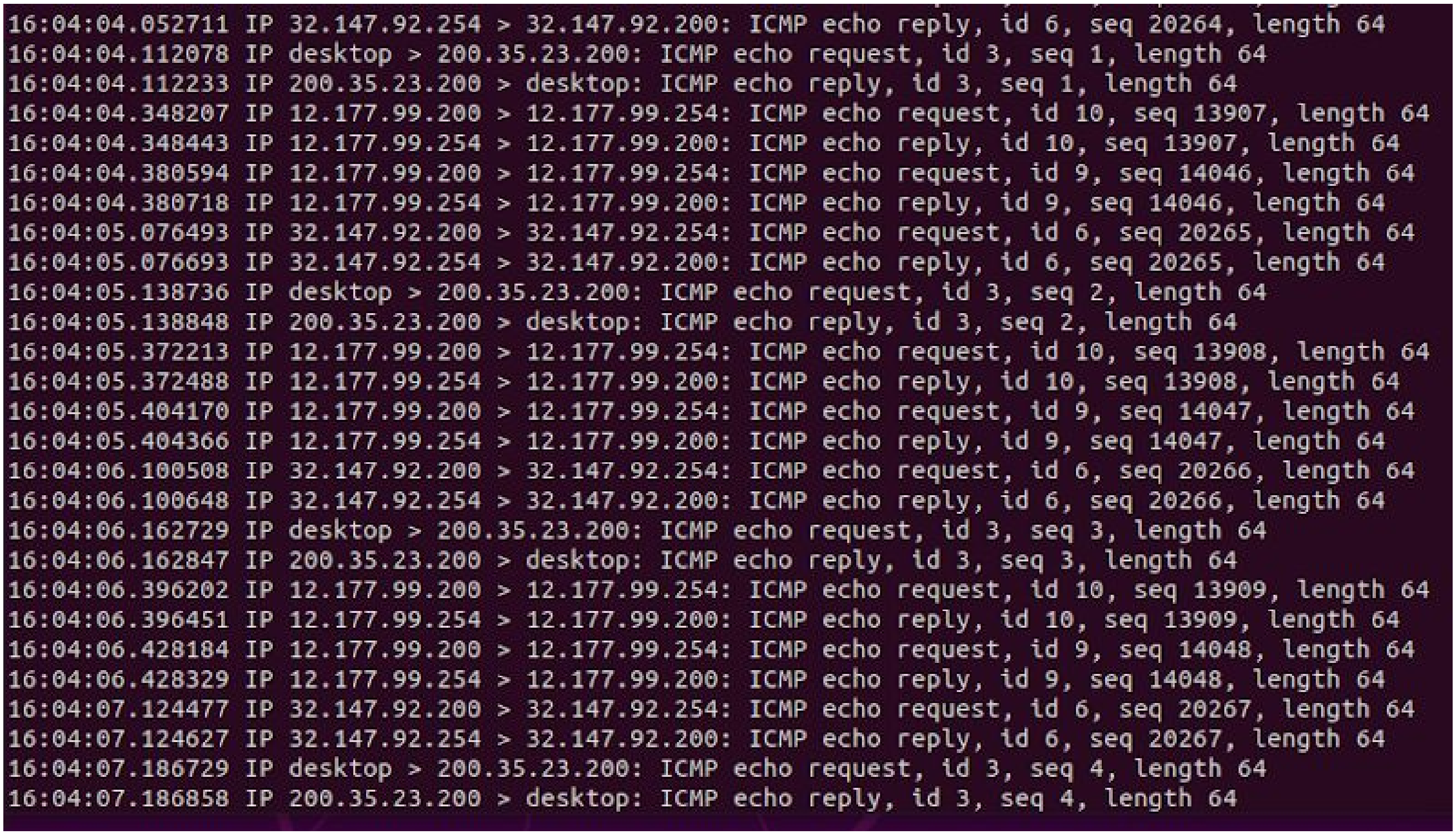


```
jboicken@ns1:~$ netstat -tl
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
jboicken@ns1:~$ netstat -tln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
jboicken@ns1:~$
```

Complete the netstat flag table

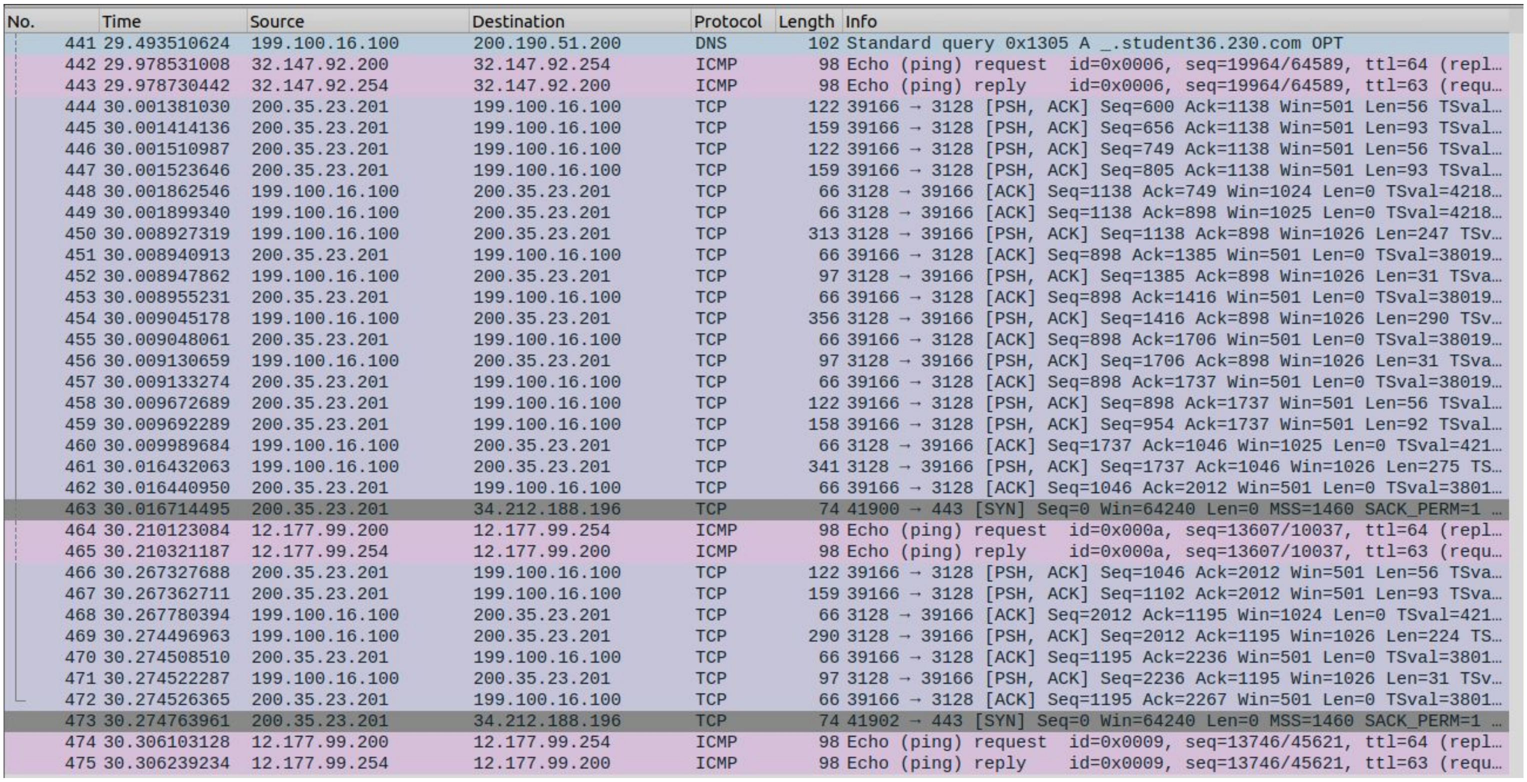
Flag	Meaning
`-t`	Tells netstat to show only connections using tcp protocol
`-l`	Tells netstat to only show the ports that are listening / open
`-n`	Shows what port number is being used instead of the protocol name

Screenshot of the captured icmp traffic using `tcpdump`



```
16:04:04.052711 IP 32.147.92.254 > 32.147.92.200: ICMP echo reply, id 6, seq 20264, length 64
16:04:04.112078 IP desktop > 200.35.23.200: ICMP echo request, id 3, seq 1, length 64
16:04:04.112233 IP 200.35.23.200 > desktop: ICMP echo reply, id 3, seq 1, length 64
16:04:04.348207 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 10, seq 13907, length 64
16:04:04.348443 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 10, seq 13907, length 64
16:04:04.380594 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 9, seq 14046, length 64
16:04:04.380718 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 9, seq 14046, length 64
16:04:05.076493 IP 32.147.92.200 > 32.147.92.254: ICMP echo request, id 6, seq 20265, length 64
16:04:05.076693 IP 32.147.92.254 > 32.147.92.200: ICMP echo reply, id 6, seq 20265, length 64
16:04:05.138736 IP desktop > 200.35.23.200: ICMP echo request, id 3, seq 2, length 64
16:04:05.138848 IP 200.35.23.200 > desktop: ICMP echo reply, id 3, seq 2, length 64
16:04:05.372213 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 10, seq 13908, length 64
16:04:05.372488 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 10, seq 13908, length 64
16:04:05.404170 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 9, seq 14047, length 64
16:04:05.404366 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 9, seq 14047, length 64
16:04:06.100508 IP 32.147.92.200 > 32.147.92.254: ICMP echo request, id 6, seq 20266, length 64
16:04:06.100648 IP 32.147.92.254 > 32.147.92.200: ICMP echo reply, id 6, seq 20266, length 64
16:04:06.162729 IP desktop > 200.35.23.200: ICMP echo request, id 3, seq 3, length 64
16:04:06.162847 IP 200.35.23.200 > desktop: ICMP echo reply, id 3, seq 3, length 64
16:04:06.396202 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 10, seq 13909, length 64
16:04:06.396451 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 10, seq 13909, length 64
16:04:06.428184 IP 12.177.99.200 > 12.177.99.254: ICMP echo request, id 9, seq 14048, length 64
16:04:06.428329 IP 12.177.99.254 > 12.177.99.200: ICMP echo reply, id 9, seq 14048, length 64
16:04:07.124477 IP 32.147.92.200 > 32.147.92.254: ICMP echo request, id 6, seq 20267, length 64
16:04:07.124627 IP 32.147.92.254 > 32.147.92.200: ICMP echo reply, id 6, seq 20267, length 64
16:04:07.186729 IP desktop > 200.35.23.200: ICMP echo request, id 3, seq 4, length 64
16:04:07.186858 IP 200.35.23.200 > desktop: ICMP echo reply, id 3, seq 4, length 64
```

Screenshot of the captured `http` traffic using Wireshark



No.	Time	Source	Destination	Protocol	Length	Info
441	29.493510624	199.100.16.100	200.190.51.200	DNS	102	Standard query 0x1305 A _.student36.230.com OPT
442	29.978531008	32.147.92.200	32.147.92.254	ICMP	98	Echo (ping) request id=0x0006, seq=19964/64589, ttl=64 (repl…
443	29.978730442	32.147.92.254	32.147.92.200	ICMP	98	Echo (ping) reply id=0x0006, seq=19964/64589, ttl=63 (requ…
444	30.001381030	200.35.23.201	199.100.16.100	TCP	122	39166 → 3128 [PSH, ACK] Seq=600 Ack=1138 Win=501 Len=56 TSval…
445	30.001414136	200.35.23.201	199.100.16.100	TCP	159	39166 → 3128 [PSH, ACK] Seq=656 Ack=1138 Win=501 Len=93 TSval…
446	30.001510987	200.35.23.201	199.100.16.100	TCP	122	39166 → 3128 [PSH, ACK] Seq=749 Ack=1138 Win=501 Len=56 TSval…
447	30.001523646	200.35.23.201	199.100.16.100	TCP	159	39166 → 3128 [PSH, ACK] Seq=805 Ack=1138 Win=501 Len=93 TSval…
448	30.001862546	199.100.16.100	200.35.23.201	TCP	66	3128 → 39166 [ACK] Seq=1138 Ack=749 Win=1024 Len=0 TSval=4218…
449	30.001899340	199.100.16.100	200.35.23.201	TCP	66	3128 → 39166 [ACK] Seq=1138 Ack=898 Win=1025 Len=0 TSval=4218…
450	30.008927319	199.100.16.100	200.35.23.201	TCP	313	3128 → 39166 [PSH, ACK] Seq=1138 Ack=898 Win=1026 Len=247 TSv…
451	30.008940913	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=898 Ack=1385 Win=501 Len=0 TSval=38019…
452	30.008947862	199.100.16.100	200.35.23.201	TCP	97	3128 → 39166 [PSH, ACK] Seq=1385 Ack=898 Win=1026 Len=31 TSva…
453	30.008955231	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=898 Ack=1416 Win=501 Len=0 TSval=38019…
454	30.009045178	199.100.16.100	200.35.23.201	TCP	356	3128 → 39166 [PSH, ACK] Seq=1416 Ack=898 Win=1026 Len=290 TSv…
455	30.009048061	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=898 Ack=1706 Win=501 Len=0 TSval=38019…
456	30.009130659	199.100.16.100	200.35.23.201	TCP	97	3128 → 39166 [PSH, ACK] Seq=1706 Ack=898 Win=1026 Len=31 TSva…
457	30.009133274	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=898 Ack=1737 Win=501 Len=0 TSval=38019…
458	30.009672689	200.35.23.201	199.100.16.100	TCP	122	39166 → 3128 [PSH, ACK] Seq=898 Ack=1737 Win=501 Len=56 TSval…
459	30.009692289	200.35.23.201	199.100.16.100	TCP	158	39166 → 3128 [PSH, ACK] Seq=954 Ack=1737 Win=501 Len=92 TSval…
460	30.009989684	199.100.16.100	200.35.23.201	TCP	66	3128 → 39166 [ACK] Seq=1737 Ack=1046 Win=1025 Len=0 TSval=421…
461	30.016432063	199.100.16.100	200.35.23.201	TCP	341	3128 → 39166 [PSH, ACK] Seq=1737 Ack=1046 Win=1026 Len=275 TS…
462	30.016440950	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=1046 Ack=2012 Win=501 Len=0 TSval=3801…
463	30.016714495	200.35.23.201	34.212.188.196	TCP	74	41900 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 …
464	30.210123084	12.177.99.200	12.177.99.254	ICMP	98	Echo (ping) request id=0x000a, seq=13607/10037, ttl=64 (repl…
465	30.210321187	12.177.99.254	12.177.99.200	ICMP	98	Echo (ping) reply id=0x000a, seq=13607/10037, ttl=63 (requ…
466	30.267327688	200.35.23.201	199.100.16.100	TCP	122	39166 → 3128 [PSH, ACK] Seq=1046 Ack=2012 Win=501 Len=56 TSva…
467	30.267362711	200.35.23.201	199.100.16.100	TCP	159	39166 → 3128 [PSH, ACK] Seq=1102 Ack=2012 Win=501 Len=93 TSva…
468	30.267780394	199.100.16.100	200.35.23.201	TCP	66	3128 → 39166 [ACK] Seq=2012 Ack=1195 Win=1024 Len=0 TSval=421…
469	30.274496963	199.100.16.100	200.35.23.201	TCP	290	3128 → 39166 [PSH, ACK] Seq=2012 Ack=1195 Win=1026 Len=224 TS…
470	30.274508510	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=1195 Ack=2236 Win=501 Len=0 TSval=3801…
471	30.274522287	199.100.16.100	200.35.23.201	TCP	97	3128 → 39166 [PSH, ACK] Seq=2236 Ack=1195 Win=1026 Len=31 TSv…
472	30.274526365	200.35.23.201	199.100.16.100	TCP	66	39166 → 3128 [ACK] Seq=1195 Ack=2267 Win=501 Len=0 TSval=3801…
473	30.274763961	200.35.23.201	34.212.188.196	TCP	74	41902 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 …
474	30.306103128	12.177.99.200	12.177.99.254	ICMP	98	Echo (ping) request id=0x0009, seq=13746/45621, ttl=64 (repl…
475	30.306239234	12.177.99.254	12.177.99.200	ICMP	98	Echo (ping) reply id=0x0009, seq=13746/45621, ttl=63 (requ…