

# Misc.

## Serving Documents via HTTP

```
while true; do nc -l -p 80 -q 1 < error.html; done
while true; do { echo -e 'HTTP/1.1 200 OK\r\n'; cat index.html; } | nc -l
8080; done
python2 -m SimpleHTTPServer 8080
python3 -m http.server 8080
twistd -n web -path .
php -S localhost:8080
```

## View markdown files in lynx

```
pandoc -s -f markdown -t html =(curl
https://raw.githubusercontent.com/cyberhouse/doc/master/README.md) | lynx -stdin
pandoc -s -f markdown -t html doc/README.md | lynx -stdin
```

## Set Host Header in curl

```
curl --header 'Host: www.example.com' 'http://localhost/index.php?id=23&L=5'
```

## Cookies and Login with curl

```
curl -d @$HOME/.credentials --cookie-jar /tmp/cjar -k
https://example.com/?do=login; curl -k --cookie /tmp/cjar --cookie-jar
/tmp/cjar https://example.com/foo|lynx -stdin
```

## Get HTTP Status Code with curl

```
curl -s -w "%{http_code}\n" http://www.cyberhouse.at -o /dev/null
# Add -L if you intend to follow location responses
```

## See File Access for a Specific Path

```
ls -ld /var/log
```

## Find Rogue Internet Gateways on your Network

```
nmap -sn 172.16.35.0/24 --script ip-forwarding --script-args="target=www.google.com"
```

## Remote Network Protocol Analyzing with tcpdump or tshark and Wireshark

```
ssh server1 tcpdump -i eth3 -U -s0 -w - 'tcp port 80' | wireshark -k -w /tmp/gw.cap -b filesize:50000 -b files:10 -i -  
ssh server1 'tshark -f "port !22" -w -' | wireshark -k -i -
```

## Create Diff with remote file

```
diff .ssh/config <(ssh trillian 'cat .ssh/config')
```

## Create cronjob programmatically

```
crontab -l > /tmp/$(whoami)-crontab  
echo '* * * * * www-data /var/www/html/typo3/cli_dispatch.phpsh scheduler'  
>> /tmp/$(whoami)-crontab  
crontab /tmp/$(whoami)-crontab  
rm /tmp/$(whoami)-crontab
```

## Forward Ports

socat TCP4-LISTEN:1234,fork TCP4:192.168.1.1:22' forwards your port 1234 to another machine's port 22. Very useful for quick NAT red

## Sysstat's sar with 24h Time Format

```
sar -o /tmp/sarlog -A 5 >/dev/null 2>&1  
LANG=C; S_TIME_FORMAT="%T; sar -f /tmp/sarlog|les
```

## Setting and Removing the Immutable Bit

```
chattr +i /etc/shadow; lsattr /etc/shadow
```

```
chattr -i /etc/shadow; lsattr /etc/shadow
```

## Add Swap Space

```
dd if=/dev/zero of=/path/to/swapfile count=2097152 # create 1 G file
mkswap /path/to/swapfile
swapon -p 32767 /path/to/swapfile # use with lowest priority possible
```

Optionally create an `/etc/fstab` entry:

```
/path/to/swapfile none swap sw,pri=32767 0 0
```

## Reset Root Password, Access to Boot Loader Required

Add `1` (after one blank space) or `S` or `Single` to the kernel boot parameter to start in single mode. Reset password with `passwd` check if SELinux is disabled if it does not work (`setenforce 0`). If single user mode does not work there is an alternative: Append `init=/bin/bash` to the boot options, remounting the root partition is necessary (`mount -o remount,rw /`).

## Cut Connections from a Specific Address

```
cutter 10.10.0.45
```

## Copy Disk with dd with Progress Bar and Compression

```
parted /dev/sda # Get the disk size first
dd if=/dev/sda | pv -s 60022480896 | pigz --fast >
/media/myExternalDrive/myBackup.img
```

## Write Image with dd with Progress Bar

```
IMAGE=archlinux-2014.04.01-dual.iso && dd bs=1m if=$IMAGE |
/opt/local/bin/pv -s $(wc -c $IMAGE) | dd of=/dev/disk1
```

## Get the Filesystem Age

```
tune2fs -l $(df -lh / | (read; awk '{print $1; exit}')) | grep -i created
```

## Strace All Processes of a Program, e.g. Apache

```
program=httpd; strace $(ps wuax|grep $program|grep -v grep|awk '{print "-p"$2}'|xargs) -s8192 -r -o /tmp/$program-$(date +%FT%T%z)
```

## Log Block Activity to Syslog (or dmesg, Depending on your Ssystem)

```
echo 1 > /proc/sys/vm/block_dump
```

## Sorting ps Output by RSS

```
ps auxww | sort -nk6  
# or  
ps auxww --sort +rss
```

## Better Readable ps

```
ps auxwwef  
# or to avoid line wraps  
ps auxef
```

## Get Buffered and Cached Read Speed of Block Device

```
hdparm -Tt /dev/sda
```

## Fill Memory Up to Given Percentage

```
stress --vm-bytes $(awk '/MemFree/{printf "%d\n", $2 * 0.9;}')<  
/proc/meminfo)k --vm-keep -m 1
```

## Using top in Batch Mode

```
top -b -i -d 10 >> /tmp/log/top
```

## Simple Port Check with Bash's Built-in /dev/tcp

```
/bin/bash -c "echo >/dev/tcp/144.76.8.201/443 && echo 'open'"
```

## Check sudo Permissions of User

```
sudo -U username -l
```

## Show OOM score

```
dstat --top-oom
```

## Transfer Files without SSH

remotely:

```
cat /usr/local/www/munin/memory-month.png | gzip -9c | uuencode /dev/stdin
```

locally:

```
pbpaste | uudecode -o /dev/stdout | gzip -cd > ~/Downloads/memory-month.png
```

## List SMB shares on macOS

```
smbutil statshares -a
```

## Get Public/External IP Address

```
dig +short myip.opendns.com @resolver1.opendns.com
```

```
curl ipinfo.io
```

```
curl ifconfig.me/all
```

```
curl icanhazip.com
```

```
curl myip.dnsomatic.com
```

```
curl whatismyip.akamai.com
```

## Add Timestamp to Output

```
vmstat 5 | awk '{system("/bin/echo -n $(/bin/date"); print ": " $0}' | tee
vmstat-$(date +%F)
```

```
vmstat 5 | awk '{now=strftime("%Y-%m-%d %T "); print now $0}'
```

## Monitor Occurrence of Event or State Change and Notify by Terminal Bell

```
while sleep 3; do curl --silent https://wiki.sysop.cafe/misc|grep sleep &&
tput bel; done
```

```
while sleep 3; do dig foo.bar.baz +short @8.8.8.8|grep 129 && tput bel; done
```

## Terminal bell character

```
printf '\a'
echo -en "\007"
tput bel
```

## Test Hosts on Servers not Available via DNS

```
curl -H "Host: example.com" 10.0.0.13
```

```
wget --header="Host: example.com" 10.0.0.13
```

## .htpasswd

```
printf "John:$(openssl passwd -crypt V3Ry)\n" >> .htpasswd # this example
uses crypt encryption
```

```
printf "Mary:$(openssl passwd -apr1 SEcRe7)\n" >> .htpasswd # this example
uses apr1 (Apache MD5) encryption
```

```
printf "Jane:$(openssl passwd -1 V3RySEcRe7)\n" >> .htpasswd # this example
uses MD5 encryption
```

```
(PASSWORD="SEcRe7PwD";SALT="$(openssl rand -base64 3)";SHA1=$(printf
"$PASSWORD$SALT" | openssl dgst -binary -sha1 | xxd -ps | sed 's###'`echo -
n $SALT | xxd -ps`"'#'| xxd -r -ps | base64);printf "Jim:{SSHA}$SHA1\n" >>
.htpasswd) # this example uses SSHA encryption
```

```
#!/usr/bin/perl
use strict;
```

```
chomp(my $filename=$ARGV[0]);
chomp(my $username=$ARGV[1]);
chomp(my $password=$ARGV[2]);

if (!$filename || !$username || !$password) {
    print "USAGE: ./crypt.pl filename username password\n\n";
} else {
    open FILE, ">>", $filename or die $!;
    print FILE $username . ":" . crypt($password, $username) . "\n";
    close FILE or die $!;
}
```

## Using Comments as Hashtags for Easier History Search

```
vim -R /var/db/mysql/percona57.err #mysql-error
```

## Getting SIGINFO with Ctrl-T

e.g. to get an progress information for a running program like dd, ping, rsync'.

## Delete Line from File

```
sed -i -e '3d' ~/.ssh/known_hosts
```

## Find and process files older than...

```
find /opt/tomcat/temp/ -type f -name '*.xlsx' -mtime +14 -exec rm {} \;
```

## Run (Cron) Script only if not already running

```
4 5 * * * /usr/local/bin/flock -n /tmp/backup.lock
/usr/local/scripts/bbackupfoo.sh
```

From:  
<https://wiki.sysop.cafe/> - [https://wiki.sysop.cafe.](https://wiki.sysop.cafe/)

Permanent link:  
<https://wiki.sysop.cafe/unix:misc>

Last update: 2019-02-27T12:04:29+0100



