TOP QUESTIONS FOR SYSADMIN INTERVIEW:
DISCLAIMER: THESE QUESTIONS AND ANSWERS AREN'T AT ALL MINE. THESE WERE SCAVANGED AROUND IN THE WEB. I HOPE IT HELPS.

BASIC:
WHAT IS LINUX AND ALSO EXPLAIN THE BASIC COMPONENTS OF LINUX?
ANSWER: LINUX IS THE MOST COMMONLY USED OPERATING SYSTEM THAT IS OPEN SOURCE AND FREE. FOR ANY COMPUTER, THE OPERATING SYSTEM ACTS AS THE BACKBONE, AND IT IS MOST IMPORTANT SOFTWARE THAT IS REQUIRED FOR ANY COMPUTER.

CONSISTS OF 3 COMPONENTS WHICH ARE:

KERNEL: LINUX IS A MONOLITHIC KERNEL THAT IS FREE AND OPEN SOURCE SOFTWARE THAT IS RESPONSIBLE FOR MANAGING HARDWARE RESOURCES FOR THE USERS.
SYSTEM LIBRARY: SYSTEM LIBRARY PLAYS A VITAL ROLE BECAUSE APPLICATION PROGRAMS ACCESS KERNELS FEATURE USING SYSTEM LIBRARY.
SYSTEM UTILITY: SYSTEM UTILITY PERFORMS SPECIFIC AND INDIVIDUAL LEVEL TASKS.
WHAT ARE THE DIFFERENCES BETWEEN UNIX AND LINUX OPERATING SYSTEM?
ANSWER: LINUX IS A UNIX CLONE, THE KERNEL OF WHICH IS CREATED BY LINUS TORVALDS. THERE ARE SO MANY DIFFERENCES BETWEEN LINUX AND UNIX OPERATING SYSTEM WHICH ARE AS FOLLOWS:

OPEN SOURCE OPERATING SYSTEM:

FREE OF COST:

COMPATIBILITY AND FLEXIBILITY:

DESCRIBE BASH.
ANSWER: BASH STANDS FOR BOURNE AGAIN SHELL. BASH IS THE UNIX SHELL FOR THE GNU OPERATING SYSTEM. SO, BASH IS THE COMMAND LANGUAGE INTERPRETER THAT HELPS YOU TO ENTER YOUR INPUT, AND THUS YOU CAN RETRIEVE INFORMATION. IN A STRAIGHTFORWARD LANGUAGE, WE CAN SAY THAT IT IS A PROGRAM THAT WILL UNDERSTAND THE DATA ENTERED BY THE USER AND EXECUTE THE COMMAND AND GIVES OUTPUT.

WHAT IS CRONTAB AND EXPLAIN ITS FUNCTIONALITY AND EXPLAIN THE FORMAT OF CRONTAB?
ANSWER: CRON IS A SCHEDULER THAT EXECUTES THE COMMANDS AT A REGULAR INTERVAL AS PER THE SPECIFIC DATE AND TIME DEFINED. WE HAVE MULTIPLE USERS IN LINUX, AND ALL THE USERS CAN HAVE THEIR CRONTAB SEPARATELY. THE CRONTABS FILES ARE SAVED AT A PARTICULAR LOCATION THAT IS /VAR/SPOOL/CRON/CRONTABS.

THERE ARE SIX FIELDS IN THE FORMAT FOR THE CRONTAB THAT IS AS BELOW:

```
<DAY_OF_THE_MONTH><MONTH_OF_THE_YEAR><DAY_OF_THE_WEEK><COMMAND/PROGRAM TO EXECUTE>
```

WHAT DO YOU UNDERSTAND BY CLI?
ANSWER: CLI IS AN ACRONYM FOR COMMAND LINE INTERFACE. WE HAVE TO PROVIDE THE INFORMATION TO THE COMPUTER SO THAT IT CAN PERFORM THE FUNCTION ACCORDINGLY. IN LINUX, CLI IS THE INTERFACE THAT PROVIDES THE USER AN INTERFACE SO THAT USER CAN TYPE THE COMMANDS AND IT COMPLETE THE TASKS. CLI IS VERY EASY TO USE, BUT IT SHOULD BE TYPED VERY PRECISELY.

WHAT IS A SWAP SPACE OR SWAP PARTITION?
ANSWER: WHEN WE HAVE INSUFFICIENT RAM SPACE IN THE SYSTEM AND WE NEED MORE RAM TO PROCESS OUR APPLICATIONS THEN LINUX ALLOWS AN EXTRA ALLOCATION OF RAM IN THE PHYSICAL HARD DISK WHICH IS CALLED A SWAP SPACE. IT IS USED TO HOLD CURRENT PROGRAMS THAT ARE CURRENTLY RUNNING IN THE SYSTEM.

DESCRIBE THE ROOT ACCOUNT.

ANSWER: THE ROOT ACCOUNT RESEMBLES AN ADMINISTRATOR ACCOUNT AND PERMITS YOU TO TAKE FULL CONTROL OF THE FRAMEWORK. HERE YOU CAN MAKE AND KEEP UP CLIENT ACCOUNTS, ALLOCATING DISTINCTIVE ACCOUNTS FOR EACH USER. IT IS THE DEFAULT ACCOUNT THAT IS CREATED EVERY TIME YOU INSTALL LINUX.

WHAT IS LILO?
ANSWER: LILO OR LINUX LOADER IS THE DEFAULT BOOT LOADER FOR LINUX. IT IS INDEPENDENT OF A SPECIFIC FILE SYSTEM AND CAN BOOT OPERATING SYSTEM FROM HARD DISKS. VARIOUS PARAMETERS SUCH AS ROOT DEVICE CAN BE SET INDEPENDENTLY USING LILO.

EXPLAIN THE IMPORTANCE OF THE GNU PROJECT?
ANSWER: THE GNU PROJECT WAS BEGUN TO MAKE A WORKING FRAMEWORK WHICH WILL BE FREE FOR CLIENTS. THE CLIENTS WOULD HAVE THE OPPORTUNITY TO RUN, SHARE, CIRCULATE, STUDY, CHANGE, AND ENHANCE OR ROLL OUT NEW IMPROVEMENTS TO THE PRODUCT.

NOWADAYS THE BIT LINUX IS EXCEPTIONALLY NORMAL. OUTSIDE THE PIECE, EVERY OTHER PIECE OF THE LINUX FRAMEWORK IS GNU. IT WAS TAKEN UNDER VARIANT 2 OF GNU AND SUBSEQUENTLY, THE NAME LINUX WAS NAMED TO GNU/LINUX.

WHAT IS THE MAXIMUM LENGTH FOR A FILENAME ALLOWED IN LINUX?
ANSWER: ANY FILENAME CAN HAVE A MOST EXTREME OF 255 CHARACTERS. THIS FARTHEST POINT DOES EXCLUDE THE PATHNAME, SO ACCORDINGLY THE WHOLE PATHNAME AND FILENAME COULD VERY MUCH SURPASS 255 CHARACTERS.

WHY IS IT A BAD IDEA TO RESTORE A DC LAST BACKED UP SEVEN MONTHS AGO?
IF YOU BACK UP A DC SEVEN MONTHS OLD, YOU COULD ENCOUNTER LINGERING OBJECTS THAT LEAD TO INCONSISTENT DATA. BACKUP FILES, AS A GENERAL RULE, SHOULDN'T BE OVER 180 DAYS OLD

EXPLAIN BRIEFLY ABOUT THREE POPULAR LINUX SHELLS.
BASH SHELL – DEFAULT SHELL IN MANY LINUX/UNIX DISTRIBUTION. HAS FEATURES LIKE

EDIT COMMAND HISTORY
SHELL FUNCTIONS AND GIVES ALIASES TO IT
UNLIMITED COMMAND HISTORY
ARRAY WITH UNLIMITED SIZE WITH INDEX.
TCSH/CSH SHELL (NORMALLY CALLED C SHELL) – TCSH IS ENHANCED C SHELL,

MORE OF C LIKE SYNTAX
AUTO-COMPLETION OF WORD AND FILENAME IS PROGRAMMABLE
SPELL CHECK
JOB CONTROL
K SHELL – IT IS CALLED KORN SHELL OR KSH.MORE THAN AN INTERACTIVE SHELL, K SHELL IS A COMPLETE, POWERFUL, HIGH-LEVEL PROGRAMMING LANGUAGE. IT HAS FEATURES LIKE

OPTIONS AND VARIABLES THAT GIVE YOU MORE WAYS TO CUSTOMIZE YOUR ENVIRONMENT.
ADVANCED SECURITY FEATURES
ADVANCED REGULAR EXPRESSIONS,- WELL-KNOWN UTILITIES LIKE GREP AND AWK.
WHAT IS RAID? WHAT IS RAID0, RAID1, RAID5, RAID10?
RAID (REDUNDANT ARRAY OF INDEPENDENT DISKS; ORIGINALLY REDUNDANT ARRAY OF INEXPENSIVE DISKS) IS A WAY OF STORING THE SAME DATA IN DIFFERENT PLACES ON MULTIPLE HARD DISKS TO PROTECT DATA IN THE CASE OF A DRIVE FAILURE. HOWEVER, NOT ALL RAID LEVELS PROVIDE REDUNDANCY

STANDARD RAID LEVELS
RAID 0**:** THIS CONFIGURATION HAS STRIPING, BUT NO REDUNDANCY OF DATA. IT OFFERS THE BEST PERFORMANCE, BUT NO FAULT TOLERANCE.

RAID 1: ALSO KNOWN AS DISK MIRRORING, THIS CONFIGURATION CONSISTS OF AT LEAST TWO

DRIVES THAT DUPLICATE THE STORAGE OF DATA. THERE IS NO STRIPING. READ PERFORMANCE IS IMPROVED SINCE EITHER DISK CAN BE READ AT THE SAME TIME. WRITE PERFORMANCE IS THE SAME AS FOR SINGLE DISK STORAGE.

RAID 5: THIS LEVEL IS BASED ON BLOCK-LEVEL STRIPING WITH PARITY. THE PARITY INFORMATION IS STRIPED ACROSS EACH DRIVE, ALLOWING THE ARRAY TO FUNCTION EVEN IF ONE DRIVE WERE TO FAIL. THE ARRAY'S ARCHITECTURE ALLOWS READ AND WRITE OPERATIONS TO SPAN MULTIPLE DRIVES. THIS RESULTS IN PERFORMANCE THAT IS USUALLY BETTER THAN THAT OF A SINGLE DRIVE, BUT NOT AS HIGH AS THAT OF A RAID 0 ARRAY. RAID 5 REQUIRES AT LEAST THREE DISKS, BUT IT IS OFTEN RECOMMENDED TO USE AT LEAST FIVE DISKS FOR PERFORMANCE REASONS.

RAID 10: COMBINING RAID 1 AND RAID 0, THIS LEVEL IS OFTEN REFERRED TO AS RAID 10, WHICH OFFERS HIGHER PERFORMANCE THAN RAID 1, BUT AT A MUCH HIGHER COST. IN RAID 1+0, THE DATA IS MIRRORED AND THE MIRRORS ARE STRIPED.

WHAT IS A LEVEL 0 BACKUP? WHAT IS AN INCREMENTAL BACKUP?
A LEVEL 0 INCREMENTAL BACKUP, WHICH IS THE BASE FOR SUBSEQUENT INCREMENTAL BACKUPS, COPIES ALL BLOCKS CONTAINING DATA, BACKING THE DATAFILE UP INTO A BACKUP SET JUST AS A FULL BACKUP WOULD.

INCREMENTAL BACKUP, ONLY STORES THE DATA THAT HAS CHANGED SINCE SOME POINT IN TIME (TYPICALLY THE PREVIOUS BACKUP)

WHAT IS VIRTUAL MEMORY?
VIRTUAL MEMORY IS A MEMORY MANAGEMENT OF AN OPERATING SYSTEM (OS) THAT USES HARDWARE AND SOFTWARE TO ALLOW A COMPUTER TO COMPENSATE FOR PHYSICAL MEMORY SHORTAGES BY TEMPORARILY TRANSFERRING DATA FROM RANDOM ACCESS MEMORY (RAM) TO DISK STORAGE. VIRTUAL ADDRESS INCREASED USING ACTIVE MEMORY IN RAM AND INACTIVE MEMORY IN HARD DISK DRIVES (HDDS) TO FORM CONTIGUOUS ADDRESSES THAT HOLD BOTH THE APPLICATION AND ITS DATA.

WHAT DOES & DISOWN AFTER A COMMAND DO?
& PUTS THE JOB IN THE BACKGROUND, THAT IS, MAKES IT BLOCK ON ATTEMPTING TO READ INPUT, AND MAKES THE SHELL NOT WAIT FOR ITS COMPLETION.
DISOWN REMOVES THE PROCESS FROM THE SHELL'S JOB CONTROL, BUT IT STILL LEAVES IT CONNECTED TO THE TERMINAL. ONE OF THE RESULTS IS THAT THE SHELL WON'T SEND IT A SIGHUP. OBVIOUSLY, IT CAN ONLY BE APPLIED TO BACKGROUND JOBS, BECAUSE YOU CANNOT ENTER IT WHEN A FOREGROUND JOB IS RUNNING.
WHAT IS THE STICKY BIT?
A STICKY BIT IS A PERMISSION BIT WHICH IS SET ON A FILE OR FOLDER, THEREBY PERMITTING ONLY THE OWNER OR ROOT USER OF THE FILE OR FOLDER TO MODIFY, RENAME OR DELETE THE CONCERNED DIRECTORY OR FILE. NO OTHER USER WOULD BE PERMITTED TO HAVE THESE PRIVILEGES ON A FILE WHICH HAS A STICKY BIT. IN UNIX-LIKE SYSTEMS, WITHOUT THE STICKY BIT ON, ANY USER CAN MODIFY, RENAME OR DELETE THE DIRECTORY OR FILE REGARDLESS OF THE OWNER OF THE FILE OR FOLDER.

WHAT DOES THE IMMUTABLE BIT DO TO A FILE?
A FILE WITH AN IMMUTABLE ATTRIBUTE CAN NOT BE:

MODIFIED
DELETED
RENAMED
NO SOFT OR HARD LINK CREATED BY ANYONE INCLUDING ROOT USER.
ONLY THE ROOT (SUPERUSER) OR A PROCESS POSSESSING THE CAP_LINUX_IMMUTABLE CAPABILITY CAN SET OR CLEAR THIS ATTRIBUTE. USE THE LSATTR COMMAND TO LIST FILE ATTRIBUTES ON A LINUX SECOND EXTENDED FILE SYSTEM THAT YOU SET WITH THE CHATTR COMMAND.

HOW TO FORCE/TRIGGER A FILE SYSTEM CHECK ON NEXT REBOOT?

THE SIMPLEST WAY TO FORCE FSCK FILESYSTEM CHECK ON A ROOT PARTITION EG.

```
/DEV/SDA1
```

IS TO CREATE AN EMPTY FILE CALLED FORCEFSCK IN THE PARTITION'S ROOT DIRECTORY.

```
# TOUCH /FORCEFSCK
```

THIS EMPTY FILE WILL TEMPORARILY OVERRIDE ANY OTHER SETTINGS AND FORCE FSCK TO CHECK THE FILESYSTEM ON THE NEXT SYSTEM REBOOT. ONCE THE FILESYSTEM IS CHECKED THE FORCEFSCK FILE WILL BE REMOVED THUS NEXT TIME YOU REBOOT YOUR FILESYSTEM WILL NOT BE CHECKED AGAIN.

WHAT IS A RUNLEVEL AND HOW TO GET THE CURRENT RUNLEVEL?

A RUNLEVEL IS ONE OF THE MODES THAT A UNIX-BASED OPERATING SYSTEM WILL RUN IN. IN LINUX KERNEL, THERE ARE 7 RUNLEVELS EXISTS, STARTING FROM 0 TO 6. THE SYSTEM CAN BE BOOTED INTO ONLY ONE RUNLEVEL AT A TIME. BY DEFAULT, A SYSTEM BOOTS EITHER TO RUNLEVEL 3 OR TO RUNLEVEL 5. RUNLEVEL 3 IS CLI, AND 5 IS GUI. THE DEFAULT RUNLEVEL IS SPECIFIED IN /ETC/INITTAB FILE IN MOST LINUX OPERATING SYSTEMS. USING RUNLEVEL, WE CAN EASILY FIND OUT WHETHER X IS RUNNING, OR NETWORK IS OPERATIONAL, AND SO ON. IN THIS BRIEF GUIDE, WE WILL TALK ABOUT HOW TO CHECK THE RUNLEVEL IN UNIX-LIKE OPERATING SYSTEMS.

HERE IS THE LIST OF RUNLEVELS IN LINUX DISTRIBUTIONS,WHICH WERE DISTRIBUTED WITH SYSV INIT AS DEFAULT SERVICE MANAGER.

0 – HALT
1 – SINGLE-USER TEXT MODE
2 – NOT USED (USER-DEFINABLE)
3 – FULL MULTI-USER TEXT MODE
4 – NOT USED (USER-DEFINABLE)
5 – FULL MULTI-USER GRAPHICAL MODE (WITH AN X-BASED LOGIN SCREEN)
6- REBOOT

TO FIND OUT THE SYSTEM RUNLEVEL, OPEN YOUR TERMINAL AND RUN THE FOLLOWING COMMAND:

```
$ RUNLEVEL
```

SAMPLE OUTPUT FOR THE ABOVE COMMAND WOULD BE:

```
N 3
```

WHAT ARE THE STEPS TO ADD A USER TO A SYSTEM WITHOUT USING USERADD/ADDUSER?

STEP-I(THE SYSTEM CREATES A DIRECTORY WITH THE NAME OF ‘USER’ IN “/HOME” DIRECTORY)
NOW WE CREATE A USER WITH A USERNAME TECHBROWN. SO START WITH THE FIRST STEP

```
[ROOT@TECHBROWN] # MKDIR /HOME/TECHBROWN
```

COPY

IT WILL CREATE A HOME DIRECTORY FOR USER TECHBROWN

STEP-II(CREATES AN ENTRY IN /ETC/PASSWD FILE)

```
[ROOT@TECHBROWN] # TOUCH /VAR/SPOOL/MAIL/TECHBROWN
```

COPY

THIS COMMAND WILL CREATE A FILE IN THE MAIL DIRECTORY SO THAT ALL MAIL’S COME TO THE USER TECHBROWN DIRECTLY STORES IN THIS FILE.

STEP-III(CREATES AN ENTRY IN /ETC/SHADOW FILE)
NOW WE CREATE AN ENTRY IN PASSWD FILE SO THAT THE GETTY SCRIPT WILL DISCOVER A INFORMATION ABOUT THE USER.

```
[ROOT@TECHBROWN] # VIM /ETC/PASSWD
```
COPY

NOW IT WILL SHOW SOME USERS INFORMATION WHICH THAT ARE PREVIOUSLY CREATED. SO NOW JUST MAKE AN ENTRY IN IT AS FOLLOWING OR SIMPLE WAY JUST COPY ONE OF THE ENTRY FROM IT. BY USING “YY(YANK OR COPY)” AND “P(PASTE)” AND THEN EDIT IT.

```
TECHBROWN:X:501:501:HELLO TECHBROWN :/HOME/GOPAL:/BIN/BASH
1 :2 :3 :4 :5: 6 : 7
```
COPY

:WQ HERE IN “/ETC/PASSWD“FILE YOU HAVE TO CREATE 7 ENTRIES. LET’S DISCUSS ABOUT IT SHORTLY. USERNAME IT INDICATES THAT THE PASSWORD IS ENCRYPTED AND STORES IN A SHADOW FILE USER ID GROUP ID COMMENT HOME DIRECTORY OF THE USER LAST ONE SHELL PROMPT YOU CAN CHECK IT THROUGH “/ETC/SHELLS” FILE.

STEP-IV(CREATES AN ENTRY IN /ETC/GROUPS FILE)
NOW WE ARE GOING TO MAKE ENTRY IN THE SHADOW FILE WHICH STORES THE INFORMATION ABOUT AN USER WITH THE ENCRYPTED PASSWORD.

```
[ROOT@TECHBROWN] # VIM /ETC/SHADOW
```
COPY

NOW MAKE ENTRY IN THIS FILE.

```
TECHBROWN:! !:16244:0:99999:7: : :
1 :2 :3 :4:5 :6:7:8:9
```
COPY

:WQ! YOU WILL SEE 9 ENTRIES PRESENTLY AVAILABLE IN THE /ETC/SHADOW FILE SO WE WILL ALSO DISCUSS ABOUT THAT. NAME OF A USER PASSWORD IN ENCRYPTED THE NUMBER OF DAYS SINCE 1 JANUARY 1970 THAT THE PASSWORD LAST CHANGED. THE NUMBER OF DAYS PERMITTED BEFORE THE PASSWORD CAN BE CHANGED. THE NUMBER OF DAYS AFTER WHICH THE PASSWORD MUST BE CHANGED. THE NUMBER OF DAYS BEFORE THE PASSWORD EXPIRES THAT THE USER IS WARNED. THE NUMBER OF DAYS AFTER THE PASSWORD EXPIRES BEFORE THE ACCOUNT IS DISABLED. THE NUMBER OF DAYS SINCE 1 JANUARY 1970 AFTER WHICH THE ACCOUNT IS DISABLED. RESERVED FOR THE FEATURE. BUT IN ABOVE WE DIDN’T EDIT THAT SO JUST DO THE FOLLOWING STEP FIRST GO TO THE SHELL PROMPT AND TYPE FOLLOWING COMMAND

```
[ROOT@TECHBROWN] # GRUB-MD5-CRYPT
PASSWORD:[TYPE YOUR PASSWORD]
RETYPE PASSWORD : [TYPE ABOVE PASSWORD AGAIN]
$1$YGGPM1$HHDEBEY0MRPKCCGYQSWQN0
```
COPY

NOW COPY AND PASTE THIS PASSWORD IN THIS SECTION.

STEP-V(CREATE A FILE FOR MAIL ADDRESS SO THAT THE MAIL COME TO THAT USER WILL BE SHOWN IN THAT FILE WHICH IS PRESENT IN “/VAR/SPOOL/MAIL/’USERNAME’”)
NOW CREATE A ENTRY IN THE /ETC/GROUPS DIRECTORY.

```
[ROOT@TECHBROWN] # VIM /ETC/GROUPS
TECHBROWN :X : 501:
1 :2 : 3 :4
```
COPY

:WQ HERE 1:2:3:4 AS USERNAME PASSWORD GROUP ID LIST OF USERS, WHICH ARE ASSOCIATED WITH THE GROUP.

STEP-VI(CREATE THE BASH PROMPTS IN ITS HOME DIRECTORY)

```
[ROOT@TECHBROWN] # TOUCH /VAR/SPOOL/MAIL/TECHBROWN
```

COPY

THIS WILL CREATE A MAIL BOX FOR THE USER FOR TECHBROWN SO THAT THE MAIL GENERATED FOR USER TECHBROWN COMES TO THIS FILE. NOW USE THIS COMMAND TO LOGIN INTO USER TECHBROWN.

```
[ROOT@TECHBROWN ~]# SU - TECHBROWN
-BASH-4.1$
```

COPY

```
[ROOT@TECHBROWN ~]#
```

COPY

THIS SHOWS THE ABOVE ERROR THAT IS A BASH ERROR. MEANS TO ENTER INTO THE USER, YOU SHOULD HAVE SOME BASH FILES INTO THE HOME DIRECTORY OF THE USER. SO DO THE FOLLOWING STEPS.

```
[ROOT@TECHBROWN ~]# CD /ETC/SKEL/
[ROOT@TECHBROWN SKEL]# CP .BASH* /HOME/TECHBROWN
[ROOT@TECHBROWN SKEL]# SU - TECHBROWN
[TECHBROWN@TECHBROWN ~]$ [YOU ARE IN USER TECHBROWN ]
```

WHAT IS MAJOR AND MINOR NUMBERS OF SPECIAL FILES?

IF YOU ISSUE THE LS -L COMMAND, YOU’LL SEE TWO NUMBERS (SEPARATED BY A COMMA) IN THE DEVICE FILE ENTRIES BEFORE THE DATE OF LAST MODIFICATION, WHERE THE FILE LENGTH NORMALLY APPEARS. THESE NUMBERS ARE THE MAJOR DEVICE NUMBER AND MINOR DEVICE NUMBER FOR THE PARTICULAR DEVICE. THE FOLLOWING LISTING SHOWS A FEW DEVICES AS THEY APPEAR ON A TYPICAL SYSTEM. THEIR MAJOR NUMBERS ARE 1, 4, 7, AND 10, WHILE THE MINORS ARE 1, 3, 5, 64, 65, AND 129.

```
 CRW-RW-RW- 1 ROOT   ROOT    1, 3   FEB 23 1999  NULL
 CRW------- 1 ROOT   ROOT   10, 1   FEB 23 1999  PSAUX
 CRW------- 1 RUBINI TTY     4, 1   AUG 16 22:22 TTY1
 CRW-RW-RW- 1 ROOT   DIALOUT 4, 64  JUN 30 11:19 TTYS0
 CRW-RW-RW- 1 ROOT   DIALOUT 4, 65  AUG 16 00:00 TTYS1
 CRW------- 1 ROOT   SYS     7, 1   FEB 23 1999  VCS1
 CRW------- 1 ROOT   SYS     7, 129 FEB 23 1999  VCSA1
 CRW-RW-RW- 1 ROOT   ROOT    1, 5   FEB 23 1999  ZERO
```

THE MAJOR NUMBER IDENTIFIES THE DRIVER ASSOCIATED WITH THE DEVICE. FOR EXAMPLE, /DEV/NULL AND /DEV/ZERO ARE BOTH MANAGED BY DRIVER 1, WHEREAS VIRTUAL CONSOLES AND SERIAL TERMINALS ARE MANAGED BY DRIVER 4;

THE MINOR NUMBER IS USED ONLY BY THE DRIVER SPECIFIED BY THE MAJOR NUMBER; OTHER PARTS OF THE KERNEL DON’T USE IT, AND MERELY PASS IT ALONG TO THE DRIVER. IT IS COMMON FOR A DRIVER TO CONTROL SEVERAL DEVICES (AS SHOWN IN THE LISTING); THE MINOR NUMBER PROVIDES A WAY FOR THE DRIVER TO DIFFERENTIATE AMONG THEM.

DESCRIBE THE MKNOD COMMAND AND WHEN YOU'D USE IT.

MKNOD WAS ORIGINALLY USED TO CREATE THE CHARACTER AND BLOCK DEVICES THAT POPULATE /DEV/. NOWADAYS SOFTWARE LIKE UDEV AUTOMATICALLY CREATES AND REMOVES DEVICE NODES ON THE VIRTUAL FILESYSTEM WHEN THE CORRESPONDING HARDWARE IS DETECTED BY THE KERNEL, BUT ORIGINALLY /DEV WAS JUST A DIRECTORY IN / THAT WAS POPULATED DURING INSTALL.

DESCRIBE A SCENARIO WHEN YOU GET A "FILESYSTEM IS FULL" ERROR, BUT 'DF' SHOWS THERE

IS FREE SPACE.
IT'S POSSIBLE THAT A PROCESS HAS OPENED A LARGE FILE WHICH HAS SINCE BEEN DELETED. YOU'LL HAVE TO KILL THAT PROCESS TO FREE UP THE SPACE. YOU MAY BE ABLE TO IDENTIFY THE PROCESS BY USING LSOF. ON LINUX DELETED YET OPEN FILES ARE KNOWN TO LSOF AND MARKED AS (DELETED) IN LSOF'S OUTPUT.

YOU CAN CHECK THIS WITH SUDO LSOF +L1

DESCRIBE A SCENARIO WHEN DELETING A FILE, BUT 'DF' NOT SHOWING THE SPACE BEING FREED.
DELETING THE FILE WON'T FREE THE SPACE UNTIL YOU DELETE THE PROCESSES THAT HAVE OPEN HANDLES AGAINST THAT FILE.

DESCRIBE HOW 'PS' WORKS.
ON LINUX, THE PS COMMAND WORKS BY READING FILES IN THE PROC FILESYSTEM THE DIRECTORY /PROC/*PID* CONTAINS VARIOUS FILES THAT PROVIDE INFORMATION ABOUT PROCESS PID. THE CONTENT OF THESE FILES IS GENERATED ON THE FLY BY THE KERNEL WHEN A PROCESS READS THEM.

WHAT HAPPENS TO A CHILD PROCESS THAT DIES AND HAS NO PARENT PROCESS TO WAIT FOR IT AND WHAT’S BAD ABOUT THIS?
IT BECOMES A ZOMBIE PROCESS.

ZOMBIE PROCESSES DON’T USE UP ANY SYSTEM RESOURCES. (ACTUALLY, EACH ONE USES A VERY TINY AMOUNT OF SYSTEM MEMORY TO STORE ITS PROCESS DESCRIPTOR.) HOWEVER, EACH ZOMBIE PROCESS RETAINS ITS PROCESS ID (PID). LINUX SYSTEMS HAVE A FINITE NUMBER OF PROCESS IDS – 32767 BY DEFAULT ON 32-BIT SYSTEMS. IF ZOMBIES ARE ACCUMULATING AT A VERY QUICK RATE – FOR EXAMPLE, IF IMPROPERLY PROGRAMMED SERVER SOFTWARE IS CREATING ZOMBIE PROCESSES UNDER LOAD — THE ENTIRE POOL OF AVAILABLE PIDS WILL EVENTUALLY BECOME ASSIGNED TO ZOMBIE PROCESSES, PREVENTING OTHER PROCESSES FROM LAUNCHING.

EXPLAIN BRIEFLY EACH ONE OF THE PROCESS STATES.
IN LINUX A PROCESS CAN BE IN A NUMBER OF STATES. IT'S EASIEST TO OBSERVE IT IN TOOLS LIKE PS OR TOP: IT'S USUALLY IN THE COLUMN NAMED S. THE DOCUMENTATION OF PS DESCRIBES THE POSSIBLE VALUES:

```
PROCESS STATE CODES
   R  RUNNING OR RUNNABLE (ON RUN QUEUE)
   D  UNINTERRUPTIBLE SLEEP (USUALLY IO)
   S  INTERRUPTIBLE SLEEP (WAITING FOR AN EVENT TO COMPLETE)
   Z  DEFUNCT/ZOMBIE, TERMINATED BUT NOT REAPED BY ITS PARENT
   T  STOPPED, EITHER BY A JOB CONTROL SIGNAL OR BECAUSE
      IT IS BEING TRACED
   [...]
```

WHICH LINUX FILE TYPES DO YOU KNOW?
LINUX FILE TYPES AND LS COMMAND IDENTIFIERS:

- : REGULAR FILE.
D : DIRECTORY.
C : CHARACTER DEVICE FILE.
B : BLOCK DEVICE FILE.
S : LOCAL SOCKET FILE.
P : NAMED PIPE.
L : SYMBOLIC LINK.

WHAT IS THE DIFFERENCE BETWEEN A PROCESS AND A THREAD? AND PARENT AND CHILD PROCESSES AFTER A FORK SYSTEM CALL?
A FORK GIVES YOU A BRAND NEW PROCESS, WHICH IS A COPY OF THE CURRENT PROCESS, WITH THE SAME CODE SEGMENTS. AS THE MEMORY IMAGE CHANGES (TYPICALLY THIS IS DUE TO DIFFERENT BEHAVIOR OF THE TWO PROCESSES) YOU GET A SEPARATION OF THE MEMORY IMAGES

(COPY ON WRITE), HOWEVER THE EXECUTABLE CODE REMAINS THE SAME. TASKS DO NOT SHARE MEMORY UNLESS THEY USE SOME INTER PROCESS COMMUNICATION (IPC) PRIMITIVE.

ONE PROCESS CAN HAVE MULTIPLE THREADS, EACH EXECUTING IN PARALLEL WITHIN THE SAME CONTEXT OF THE PROCESS. MEMORY AND OTHER RESOURCES ARE SHARED AMONG THREADS, THEREFORE SHARED DATA MUST BE ACCESSED THROUGH SOME PRIMITIVE AND SYNCHRONIZATION OBJECTS THAT ALLOW YOU TO AVOID DATA CORRUPTION.

ALL THE PROCESSES IN OPERATING SYSTEM ARE CREATED WHEN A PROCESS EXECUTES THE FORK() SYSTEM CALL EXCEPT THE STARTUP PROCESS. THE PROCESS THAT USED THE FORK() SYSTEM CALL IS THE PARENT PROCESS. IN OTHER WORDS, A PARENT PROCESS IS ONE THAT CREATES A CHILD PROCESS. A PARENT PROCESS MAY HAVE MULTIPLE CHILD PROCESSES BUT A CHILD PROCESS ONLY ONE PARENT PROCESS.

ON THE SUCCESS OF A FORK() SYSTEM CALL, THE PID OF THE CHILD PROCESS IS RETURNED TO THE PARENT PROCESS AND 0 IS RETURNED TO THE CHILD PROCESS. ON THE FAILURE OF A FORK() SYSTEM CALL, -1 IS RETURNED TO THE PARENT PROCESS AND A CHILD PROCESS IS NOT CREATED.

A CHILD PROCESS IS A PROCESS CREATED BY A PARENT PROCESS IN OPERATING SYSTEM USING A FORK() SYSTEM CALL. A CHILD PROCESS MAY ALSO BE CALLED A SUBPROCESS OR A SUBTASK.

A CHILD PROCESS IS CREATED AS ITS PARENT PROCESS’S COPY AND INHERITS MOST OF ITS ATTRIBUTES. IF A CHILD PROCESS HAS NO PARENT PROCESS, IT WAS CREATED DIRECTLY BY THE KERNEL.

IF A CHILD PROCESS EXITS OR IS INTERRUPTED, THEN A SIGCHLD SIGNAL IS SEND TO THE PARENT PROCESS.

WHAT IS THE DIFFERENCE BETWEEN EXEC AND FORK?
FORK STARTS A NEW PROCESS WHICH IS A COPY OF THE ONE THAT CALLS IT, WHILE EXEC REPLACES THE CURRENT PROCESS IMAGE WITH ANOTHER (DIFFERENT) ONE.
BOTH PARENT AND CHILD PROCESSES ARE EXECUTED SIMULTANEOUSLY IN CASE OF FORK() WHILE CONTROL NEVER RETURNS TO THE ORIGINAL PROGRAM UNLESS THERE IS AN EXEC() ERROR.
WHAT IS "NOHUP" USED FOR?
NOHUP DISCONNECTS THE PROCESS FROM THE TERMINAL, REDIRECTS ITS OUTPUT TO NOHUP.OUT AND SHIELDS IT FROM SIGHUP. ONE OF THE EFFECTS (THE NAMING ONE) IS THAT THE PROCESS WON'T RECEIVE ANY SENT SIGHUP. IT IS COMPLETELY INDEPENDENT FROM JOB CONTROL AND COULD IN PRINCIPLE BE USED ALSO FOR FOREGROUND JOBS (ALTHOUGH THAT'S NOT VERY USEFUL).
HOW CAN YOU GET HOST, CHANNEL, ID, LUN OF SCSI DISK?

```
$ CAT /PROC/SCSI/SCSI
HOST: SCSI2 CHANNEL: 00 ID: 00 LUN: 29
  VENDOR: EMC      MODEL: SYMMETRIX
$ LS -LD /SYS/BLOCK/SD*/DEVICE
LRWXRWXRWX 1 ROOT ROOT 0 OCT  4 12:12 /SYS/BLOCK/SDAZ/DEVICE ->
../../DEVICES/PCI0000:20/0000:20:02.0/0000:27:00.0/HOST2/RPORT-2:0-0/
TARGET2:0:0/2:0:0:29
LRWXRWXRWX 1 ROOT ROOT 0 OCT  4 12:12 /SYS/BLOCK/SDBI/DEVICE ->
../../DEVICES/PCI0000:20/0000:20:02.2/0000:24:00.0/HOST3/RPORT-3:0-0/
TARGET3:0:0/3:0:0:29
```

HOW CAN YOU LIMIT PROCESS MEMORY USAGE?
ULIMIT ALLOWS YOU TO LIMIT THE RESOURCES THAT A PROCESS CAN USE

CAN YOU EXPLAIN TO ME THE DIFFERENCE BETWEEN BLOCK BASED, AND OBJECT BASED STORAGE?
OBJECT STORAGE (ALSO REFERRED TO AS OBJECT-BASED STORAGE) IS A GENERAL TERM THAT REFERS TO THE WAY IN WHICH WE ORGANIZE AND WORK WITH UNITS OF STORAGE, CALLED OBJECTS. EVERY OBJECT CONTAINS THREE THINGS:

THE DATA ITSELF. THE DATA CAN BE ANYTHING YOU WANT TO STORE, FROM A FAMILY PHOTO TO A 400,000-PAGE MANUAL FOR ASSEMBLING AN AIRCRAFT.
AN EXPANDABLE AMOUNT OF METADATA. THE METADATA IS DEFINED BY WHOEVER CREATES THE OBJECT STORAGE; IT CONTAINS CONTEXTUAL INFORMATION ABOUT WHAT THE DATA IS, WHAT IT SHOULD BE USED FOR, ITS CONFIDENTIALITY, OR ANYTHING ELSE THAT IS RELEVANT TO THE WAY IN WHICH THE DATA IS USED.
A GLOBALLY UNIQUE IDENTIFIER. THE IDENTIFIER IS AN ADDRESS GIVEN TO THE OBJECT IN ORDER FOR THE OBJECT TO BE FOUND OVER A DISTRIBUTED SYSTEM. THIS WAY, IT'S POSSIBLE TO FIND THE DATA WITHOUT HAVING TO KNOW THE PHYSICAL LOCATION OF THE DATA (WHICH COULD EXIST WITHIN DIFFERENT PARTS OF A DATA CENTER OR DIFFERENT PARTS OF THE WORLD).
WITH BLOCK STORAGE, FILES ARE SPLIT INTO EVENLY SIZED BLOCKS OF DATA, EACH WITH ITS OWN ADDRESS BUT WITH NO ADDITIONAL INFORMATION (METADATA) TO PROVIDE MORE CONTEXT FOR WHAT THAT BLOCK OF DATA IS. YOU'RE LIKELY TO ENCOUNTER BLOCK STORAGE IN THE MAJORITY OF ENTERPRISE WORKLOADS; IT HAS A WIDE VARIETY OF USES (AS SEEN BY THE RISE IN POPULARITY OF SAN ARRAYS).

OBJECT STORAGE, BY CONTRAST, DOESN'T SPLIT FILES UP INTO RAW BLOCKS OF DATA. THERE IS NO LIMIT ON THE TYPE OR AMOUNT OF METADATA, WHICH MAKES OBJECT STORAGE POWERFUL AND CUSTOMIZABLE.

GIVE US A BRIEF RUNDOWN OF YOUR TROUBLESHOOTING PROCESS.
ADVANCED:
EXPLAIN NETWORK BONDING AND ALSO EXPLAIN THE DIFFERENT TYPES OF NETWORK BONDING?
ANSWER: NETWORK BONDING AS THE NAME IMPLIES THAT IT IS THE PROCESS OF BONDING OR JOINING TWO OR MORE THAN TWO NETWORK INTERFACES TO CREATE ONE INTERFACE. IT HELPS IN IMPROVING THE NETWORK THROUGHPUT, BANDWIDTH, REDUNDANCY, LOAD BALANCING AS IN CASE ANY OF THE INTERFACES IS DOWN; THE OTHER ONE WILL CONTINUE TO WORK. SEVERAL TYPES OF NETWORK BONDING ARE AVAILABLE THAT ARE BASED ON THE KIND OF BONDING METHOD.

BELOW ARE THE DIFFERENT BONDING TYPES IN LINUX:

BALANCE-RR OR MODE 0 – THIS IS THE DEFAULT MODE OF NETWORK BONDING THAT WORKS ON THE ROUND-ROBIN POLICY THAT MEANS FROM THE FIRST SLAVE TO THE LAST, AND IT IS USED FOR FAULT TOLERANCE AND LOAD BALANCING.
ACTIVE-BACKUP OR MODE 1 – THIS TYPE OF NETWORK BONDING WORKS ON THE ACTIVE-BACKUP POLICY THAT MEANS ONLY ONE SLAVE WILL BE ACTIVE AND OTHER WILL WORK JUST WHEN ANOTHER SLAVE FAILS. THIS MODE IS ALSO USED FOR FAULT TOLERANCE.
BALANCE-XOR OR MODE 2 –THIS TYPE OF NETWORK BONDING SETS AN EXCLUSIVE OR MODE THAT MEANS SOURCE MAC ADDRESS IS XOR'D WITH THE DESTINATION ADDRESS, AND THUS IT PROVIDES FAULT TOLERANCE AND LOAD BALANCING.
BROADCAST OR MODE 3 –THIS MODE SETS A BROADCAST MODE TO PROVIDE FAULT TOLERANCE, AND IT SHOULD BE USED FOR PARTICULAR PURPOSES. IN THIS TYPE OF NETWORK BONDING, ALL TRANSMISSIONS ARE SENT TO ALL SLAVE INTERFACES.
802.3AD OR MODE 4 –THIS MODE WILL CREATE THE AGGREGATION GROUPS, AND ALL THE GROUPS WILL SHARE THE SAME SPEED. FOR THIS, MODE SETS AN IEEE 802.3AD DYNAMIC LINK AGGREGATION MODE. IT IS DONE BY PARTICULAR SWITCH SUPPORT THAT SUPPORTS IEEE 802.3AD DYNAMIC LINK.
BALANCE-TLB OR MODE 5 –THIS MODE SETS A TRANSMIT LOAD BALANCING MODE FOR FAULT TOLERANCE AND LOAD BALANCING AND DOES NOT REQUIRE ANY SWITCH SUPPORT.
BALANCE-ALB OR MODE 6 –THIS MODE SETS AN ACTIVE LOAD BALANCING TO ACHIEVE FAULT TOLERANCE AND LOAD BALANCING.
WHAT IS THE SIMILARITY AND DIFFERENCE BETWEEN CRON AND ANACRON? WHICH ONE WOULD YOU PREFER TO USE?
ANSWER: HERE WE ARE GOING TO DISCUSS THE SIMILARITY AND THE DIFFERENCES BETWEEN CRON AND ANACRON. SO, LET'S START WITH THE ANALOGY:

CRON AND ANACRON ARE USED TO SCHEDULE THE TASKS IN CRON JOBS. BOTH OF THESE ARE THE

DAEMONS THAT ARE USED TO SCHEDULE THE EXECUTION OF COMMANDS OR TASKS AS PER THE INFORMATION PROVIDED BY THE USER.

DIFFERENCES BETWEEN CRON AND ANACRON:

ONE OF THE MAIN DIFFERENCE BETWEEN CRON AND ANACRON JOBS IS THAT CRON WORKS ON THE SYSTEM THAT ARE RUNNING CONTINUOUSLY THAT MEANS IT IS DESIGNED FOR THE SYSTEM THAT IS RUNNING24*7. WHILE ANACRON IS USED FOR THE SYSTEMS THAT ARE NOT RUNNING CONTINUOUSLY.

OTHER DIFFERENCE BETWEEN THE TWO IS CRON JOBS CAN RUN EVERY MINUTE, BUT ANACRON JOBS CAN BE RUN ONLY ONCE A DAY.

ANY NORMAL USER CAN DO THE SCHEDULING OF CRON JOBS, BUT THE SCHEDULING OF ANACRON JOBS CAN BE DONE BY THE SUPERUSER ONLY.

CRON SHOULD BE USED WHEN YOU NEED TO EXECUTE THE JOB AT A SPECIFIC TIME AS PER THE GIVEN TIME IN CRON, BUT ANACRON SHOULD BE USED IN WHEN THERE IS NO ANY RESTRICTION FOR THE TIMING AND CAN BE EXECUTED AT ANY TIME.

IF WE THINK ABOUT WHICH ONE IS IDEAL FOR SERVERS OR DESKTOPS, THEN CRON SHOULD BE USED FOR SERVERS WHILE ANACRON SHOULD BE USED FOR DESKTOPS OR LAPTOPS.

WHAT IS THE ISSUE BEHIND GETTING AN ERROR “FILESYSTEM IS FULL” WHILE THERE IS SPACE AVAILABLE WHEN YOU CHECK IT THROUGH “DF” COMMAND? HOW WILL YOU RECTIFY THIS PROBLEM?

ANSWER: WHEN ALL THE INODES ARE CONSUMED THEN EVEN THOUGH YOU HAVE FREE SPACE, YOU WILL GET THE ERROR THAT FILESYSTEM IS FULL. SO, TO CHECK WHETHER THERE IS SPACE AVAILABLE, WE HAVE TO USE THE COMMAND DF –I. SOMETIMES, IT MAY HAPPEN FILE SYSTEM OR STORAGE UNIT CONTAINS THE SUBSTANTIAL NUMBER OF SMALL FILES, AND EACH OF THE FILES TAKES 128 BYTES OF THE INODE STRUCTURE THEN INODE STRUCTURE FILLS UP, AND WE WILL NOT BE ABLE TO COPY ANY MORE FILE TO THE DISK. SO, TO RECTIFY THE PROBLEM, YOU NEED TO FREE THE SPACE IN INODE STORAGE, AND YOU WILL BE ABLE TO SAVE MORE FILES.

WHERE IS PASSWORD FILE LOCATED IN LINUX AND HOW CAN YOU IMPROVE THE SECURITY OF PASSWORD FILE?

ANSWER: THIS IS AN IMPORTANT QUESTION THAT IS GENERALLY ASKED BY THE INTERVIEWERS. USER INFORMATION ALONG WITH THE PASSWORDS IN LINUX IS STORED IN/ETC/PASSWD THAT IS A COMPATIBLE FORMAT. BUT THIS FILE IS USED TO GET THE USER INFORMATION BY SEVERAL TOOLS. HERE, SECURITY IS AT RISK. SO, WE HAVE TO MAKE IT SECURED.

TO IMPROVE THE SECURITY OF THE PASSWORD FILE, INSTEAD OF USING A COMPATIBLE FORMAT WE CAN USE SHADOW PASSWORD FORMAT. SO, IN SHADOW PASSWORD FORMAT, THE PASSWORD WILL BE STORED AS SINGLE “X” CHARACTER WHICH IS NOT THE SAME FILE (/ETC/PASSWD). THIS INFORMATION IS STORED IN ANOTHER FILE INSTEAD WITH A FILE NAME /ETC/SHADOW. SO, TO ENHANCE THE SECURITY, THE FILE IS MADE WORD READABLE AND ALSO, THIS FILE IS READABLE ONLY BY THE ROOT USER. THUS SECURITY RISKS ARE OVERCOME TO A GREAT EXTENT BY USING THE SHADOW PASSWORD FORMAT.

WHAT IS KEY-BASED AUTHENTICATION? EXPLAIN.

ANSWER: THERE ARE VARIOUS METHODS TO ENTER INTO THE SERVERS. ONE OF THE WAYS TO LOG IN IS USING PASSWORD-BASED AUTHENTICATION, BUT THAT IS NOT SECURE. SO, WE NEED A METHOD THAT IS SECURED.

ONE OF THE WAYS TO ACHIEVE THE SECURITY IS TO USE KEY-BASED AUTHENTICATION. TO USE THIS TYPE OF AUTHENTICATION, WE HAVE TO DISABLE THE PASSWORD-BASED AUTHENTICATION. SO, THERE IS A PROCEDURE TO SET UP THIS AUTHENTICATION WHICH IS AS FOLLOWS:

WE HAVE TO GET THE SSH KEY PAIR USING BELOW COMMAND:

```
$ SSH-KEYGEN -T RSA
```

IT WILL GENERATE THE PUBLIC/PRIVATE RSA KEY PAIR.

ENTER FILE WHERE YOU WANT TO SAVE THIS GENERATED KEY (/HOME/USERNAME/.SSH/ID_RSA):

IT WILL PROMPT YOU FOR THE SAME LOCATION, I.E. ~/.SSH/ID_RSA FOR THE KEY PAIR. PRESS ENTER IF YOU WANT TO CONFIRM THE SAME LOCATION. ELSE, IF YOU WANT TO PROVIDE ANY OTHER LOCATION, ENTER THAT AND CONFIRM THE SAME.

NOW COPY ~/.SSH/ID_RSA.PUB INTO THE ~/.SSH/AUTHORIZED_KEYS THAT WILL BE LOCATED WHERE YOU HAVE TO CONNECT.

NOW, WE HAVE TO PROVIDE THE PERMISSIONS TO THE FILE AS PER BELOW COMMAND:

```
$ CHMOD 600 ~/.SSH/AUTHORIZED_KEYS
```

NOW TRY TO SSHTHE MACHINE YOU WANT TO CONNECT, AND YOU WILL SEE THAT YOU ARE ABLE TO LOGIN TO THE MACHINE WITHOUT A PASSWORD.

IF YOU ARE CONFIRMED THAT KEY-BASED AUTHENTICATION IS WORKING FINE, DISABLE THE PASSWORD-BASED AUTHENTICATION.

GO TO THE PATH /ETC/ SSH/SSHD_CONFIG

SET THE FOLLOWING PROPERTY AS NO.

```
PASSWORDAUTHENTICATION NO
```

MENTION THE STEPS TO FIND OUT THE MEMORY USAGE BY LINUX.
ANSWER: YOU HAVE TO ENTER A COMMAND IN THE LINUX SHELL CALLED “CONCATENATE” TO FIND OUT THE MEMORY USAGE BY LINUX.

SYNTAX: CAT/PROC/MEMINFO.

WHEN YOU WILL ENTER THIS COMMAND THEN YOU WILL SEE A LIST OF MEMORY USAGE LIKE TOTAL MEMORY, FREE MEMORY, CACHE MEMORY, AND MANY OTHER MEMORY USAGES BY LINUX.
OTHER COMMANDS USED IN LINUX ARE:

```
$ FREE –M // THIS IS THE SIMPLEST COMMAND WHERE IT WILL SHOW THE MEMORY USAGE IN MB.
$ VMSTAT –S //THIS COMMAND GIVES A REPORT ON VIRTUAL MEMORY STATISTICS
TOP // THIS COMMAND CHECKS THE USAGE OF MEMORY AND CPU USAGE
HTOP // SIMILAR LIKE TOP COMMAND
```

WHAT DO YOU MEAN BY AN EXT3 FILE SYSTEM?
ANSWER: THIS IS ONE OF THE TOP LINUX INTERVIEW QUESTIONS ASKED IN THE LINUX INTERVIEW. IT CAN BE ANSWERED IN THE FOLLOWING MANNER. EXT3 FILE SYSTEM IS AN UPGRADED VERSION OF EXT2 AND IT ALSO SUPPORTS JOURNALING. WHEN AN UNCLEAN SHUTDOWN IS PERFORMED EXT2 FILE SYSTEM PERFORMS A CHECK ON THE MACHINE FOR ERRORS WHICH IS A LONG PROCESS BUT IT IS NOT SO IN CASE OF THE EXT3 FILE SYSTEM.

IN CASE OF A HARDWARE FAILURE, AN EXT3 CONSISTENCY CHECK WILL OCCUR WITHOUT ANY PAUSE. THE TIME OF THE RECOVERY OF THE FILE SYSTEM IS INDEPENDENT OF THE NUMBER OF FILES. THE TIME IS DEPENDENT ON THE SIZE OF THE JOURNAL WHICH ONLY TAKES A SECOND WHICH DEPENDS ON THE SPEED OF THE HARDWARE.

WHAT IS THE LEVEL OF SECURITY THAT LINUX PROVIDES IN COMPARISON TO OTHER OPERATING SYSTEMS?
ANSWER: IF AN OPERATING SYSTEM IS NOT SECURE THEN IT IS NOT SUCCESSFUL. IN COMPARISON TO OTHER OPERATING SYSTEMS, LINUX IS THE MOST SECURE OPERATING SYSTEM AS IT CONSISTS OF PLUGGABLE AUTHENTICATION MODULES. A SECURE LAYER IS CREATED BETWEEN THE AUTHENTICATION PROCESS AND APPLICATIONS. IT IS BECAUSE OF PAM ONLY BY WHICH AN ADMIN CAN GIVE ACCESS TO OTHER USERS TO LOG INTO THE SYSTEM. YOU CAN FIND THE CONFIGURATION OF PAM APPLICATIONS IN THE “/ETC/PAM.D” OR “/ETC/PAM.CONF” DIRECTORY.

WHAT ARE SOFT LINKS? DESCRIBE SOME OF THE FEATURES OF SOFT LINKS.
ANSWER: SOFT LINKS OR SYMBOLIC LINK OR SYMLINK ARE SPECIAL FILES WHICH ARE USED AS A REFERENCE FOR ANOTHER DIRECTORY. SOME FEATURES OF SOFTLINKS ARE:

THEY HAVE A DIFFERENT INODE NUMBER WITH RESPECT TO SOURCE FILES OR ORIGINAL FILES.
IF IN CASE THE ORIGINAL FILE IS DELETED THEN A SOFT LINK OF THAT FILE IS USELESS.
WE CANNOT UPDATE A SOFT LINK.
SOFT LINKS ARE USED TO CREATE LINKS BETWEEN DIRECTORIES.
SOFT LINKS ARE INDEPENDENT OF FILE SYSTEM BOUNDARIES.
EXPLAIN INODE IN LINUX.
ANSWER: INODE IS A STRUCTURE WHICH ACTS AS AN IDENTITY FOR ALL FILES AND OBJECTS. TYPE A COMMAND IN THE SHELL “LS -I”. THE NUMBERS WHICH ARE DISPLAYED AT THE ADJACENT OF FILES AND FOLDERS, THESE ARE INODE NUMBERS WHICH ARE ASSIGNED TO EACH FILE THAT CONTAINS INFORMATION ABOUT THE FILE. THE SYSTEM USES THIS NUMBER TO IDENTIFY THE FILE. INFORMATION LIKE THE SIZE OF THE FILE, WHEN THE FILE WAS MODIFIED ETC IS CONTAINED IN AN INODE NUMBER.

THE QUESTIONS BASED ON INODE IS THE MOST COMMON LINUX INTERVIEW QUESTION YOU MAY COME ACROSS IN YOUR LINUX INTERVIEW. SO, READ WELL AND GET ENOUGH KNOWLEDGE EVEN IF YOU ARE NOT FAMILIAR WITH IT AND GET READY WITH THE ANSWER.

WHAT IS THE ROUTING TABLE IN LINUX?
ANSWER: THE ROUTING TABLE IS A METHOD IN WHICH HOW ALL THE NETWORKS AND DEVICES ARE INTERCONNECTED WITH EACH OTHER TO EFFICIENTLY ESTABLISH COMMUNICATION WITH EACH OTHER.

WHAT IS PUPPET?
ANSWER: THE PUPPET IS OPEN SOURCE SOFTWARE WHICH IS USED FOR SOFTWARE CONFIGURATION MANAGEMENT THAT RUNS ON SYSTEMS SIMILAR TO THAT OF UNIX. IT IS SECURE AND SCALABLE TO USE. IT PROVIDES AUTOMATION FEATURES IN DEVOPS AND CLOUD ENVIRONMENT.

WHAT IS AUTOMOUNTING IN LINUX?
ANSWER: THE AUTOMOUNTING IS A PROCESS OF AUTOMATICALLY MOUNTING ALL THE PARTITIONS ON A HARD DISK ON A LINUX OR UNIX SYSTEM WHILE BOOTING THE SYSTEM. FSTAB PROPERTY CAN BE USED TO AUTOMOUNT THE HARD DRIVES ON LINUX.

LIST THE FIELDS IN /ETC/PASSWD FILE.
ANSWER: THE FIELDS THAT ARE PRESENT IN /ETC/PASSWD FILE ARE USERNAME, PASSWORD, USER ID, GROUP ID, COMMENTS, HOMEDIR AND LOGINSHELL. THE /ETC/PASSWD FILE HAS CONTENTS AS BELOW:

```
REDHAT  500:500:REDHAT USER:/HOME/REDHAT:/BIN/BASH

MSSM  501:501:ANOTHER USER:/HOME/MSSM:/BIN/BASH
```

– “X” IN THE PASSWORD COLUMN INDICATES THAT THE ENCRYPTED PASSWORD IS STORED IN /ETC/SHADOW FILE.

EXPLAIN EACH SYSTEM CALL USED FOR PROCESS MANAGEMENT IN LINUX.
ANSWER: THIS IS THE MOST POPULAR LINUX SYSTEM ADMINISTRATION INTERVIEW QUESTIONS ASKED IN AN INTERVIEW. THE SYSTEM CALLS THAT ARE USED FOR PROCESS MANAGEMENT ARE AS FOLLOWS:

FORK(): THIS IS USED TO CREATE A NEW PROCESS FROM AN EXISTING ONE.

EXEC(): THIS IS USED TO EXECUTE A NEW PROGRAM.

WAIT(): THIS IS USED TO WAIT UNTIL THE GIVEN PROCESS FINISHES THE EXECUTION.

EXIT(): THIS IS USED TO EXIT FROM THE PROCESS.

GETPID(): THIS HELPS IN GETTING THE UNIQUE PROCESS ID OF A PARTICULAR PROCESS.

GETPPID(): THIS HELPS IN GETTING A PARENT PROCESS UNIQUE ID.

NICE(): THIS IS USED TO BIAS THE EXISTING PROPERTY OF THE PROCESS.

EXPLAIN THE STEPS TO INCREASE THE SIZE OF THE LVM PARTITION.
ANSWER: THE STEPS THAT NEED TO BE FOLLOWED TO INCREASE THE SIZE OF THE LVM PARTITION ARE AS FOLLOWS:

RUN THE BELOW COMMAND: LVEXTEND -L +500M /DEV/.

ONCE THIS IS DONE WE CAN INCREASE THE SIZE OF THE LVM PARTITION BY 500MB. A USER CAN CHECK THE SIZE OF THE PARTITION BY USING ‘DF -H’ COMMAND. THE RESIZING CAN BE DONE BY RESIZE2FS /DEV/.

LET US MOVE TO THE NEXT LINUX SYSTEM ADMINISTRATION INTERVIEW QUESTIONS.

WHICH UTILITY CAN BE USED TO CREATE THE PARTITION FROM A RAW DISK?
ANSWER: TO CREATE A PARTITION FROM A RAW DISK THE UTILITY THAT IS USED IS FDISK UTILITY. TO CREATE A PARTITION YOU CAN FOLLOW THE BELOW STEPS:

RUN THIS COMMAND: FDISK /DEV/HD* (IDE) OR /DEV/SD* (SCSI).
TYPE N TO CREATE A NEW PARTITION.
ONCE A PARTITION IS CREATED THEN YOU CAN WRITE THE CHANGES TO THIS PARTITION TABLE. TO WRITE THESE CHANGES TYPE W.
. EXPLAIN THE STEPS TO CREATE A NEW USER AND SET A PASSWORD FOR THE USER FROM A SHELL PROMPT IN LINUX.
ANSWER:

TO CREATE A NEW USER ACCOUNT FROM SHELL PROMPT FOLLOWING STEPS ARE TO BE PERFORMED:

FIRSTLY LOGIN AS A ROOT USER IF YOU ARE NOT LOGGED IN AS ROOT USE SU – COMMAND.
ENTER THE ROOT PASSWORD.
THE COMMAND TO ADD A NEW USER IS USERADD COMMAND AND CAN BE USED IN LINUX. USE THIS COMMAND AND THEN TYPE THE USERNAME YOU WOULD LIKE TO CREATE.
EG: USERADD SUE

ONCE A USER IS CREATED TO SET THE PASSWORD TO FOLLOW BELOW STEPS:

TO SET A PASSWORD FOR USER SUE TYPE COMMAND: PASSWD SUE.
IT WILL PROMPT THE USER TO ENTER A NEW PASSWORD.
ONCE THIS IS DONE IT WILL ALSO ASK THE USER TO RETYPE PASSWORD THEREBY SETTING UP THE PASSWORD FOR THE USER.
WHAT IS A TUNNEL AND HOW YOU CAN BYPASS A HTTP PROXY?
HTTP TUNNELING (HA OUTROS TIPOS TIPO SSH TUNNEL....) IS THE PROCESS IN WHICH COMMUNICATIONS ARE ENCAPSULATED BY USING HTTP PROTOCOL. AN HTTP TUNNEL IS OFTEN USED FOR NETWORK LOCATIONS WHICH HAVE RESTRICTED CONNECTIVITY OR ARE BEHIND FIREWALLS OR PROXY SERVERS.

THE SERVER RUNS OUTSIDE THE PROTECTED NETWORK AND ACTS AS A SPECIAL HTTP SERVER. THE CLIENT PROGRAM IS RUN ON A COMPUTER INSIDE THE PROTECTED NETWORK. WHENEVER ANY NETWORK TRAFFIC IS PASSED TO THE CLIENT, IT REPACKAGES IT AS AN HTTP REQUEST AND RELAYS IT TO THE OUTSIDE SERVER, WHICH EXTRACTS AND EXECUTES THE ORIGINAL NETWORK REQUEST FOR THE CLIENT. THE RESPONSE TO THE REQUEST, WHICH WAS SENT TO THE SERVER IS REPACKAGED AS AN HTTP RESPONSE AND RELAYED BACK TO THE CLIENT. SINCE ALL TRAFFIC

IS ENCAPSULATED INSIDE NORMAL GET AND POST REQUESTS AND RESPONSES, THIS APPROACH WORKS THROUGH MOST PROXIES AND FIREWALLS.

WHAT IS THE DIFFERENCE BETWEEN IDS AND IPS?
AN INTRUSION DETECTION SYSTEM (IDS) IS A DEVICE OR SOFTWARE APPLICATION THAT MONITORS A NETWORK OR SYSTEMS FOR MALICIOUS ACTIVITY OR POLICY VIOLATIONS. ANY MALICIOUS ACTIVITY OR VIOLATION IS TYPICALLY REPORTED EITHER TO AN ADMINISTRATOR OR COLLECTED CENTRALLY USING A SECURITY INFORMATION AND EVENT MANAGEMENT (SIEM) SYSTEM. A SIEM SYSTEM COMBINES OUTPUTS FROM MULTIPLE SOURCES, AND USES ALARM FILTERING TECHNIQUES TO DISTINGUISH MALICIOUS ACTIVITY FROM FALSE ALARMS.

INTRUSION PREVENTION SYSTEMS (IPS), ALSO KNOWN AS INTRUSION DETECTION AND PREVENTION SYSTEMS (IDPS), ARE NETWORK SECURITY APPLIANCES THAT MONITOR NETWORK OR SYSTEM ACTIVITIES FOR MALICIOUS ACTIVITY. THE MAIN FUNCTIONS OF INTRUSION PREVENTION SYSTEMS ARE TO IDENTIFY MALICIOUS ACTIVITY, LOG INFORMATION ABOUT THIS ACTIVITY, REPORT IT AND ATTEMPT TO BLOCK OR STOP IT

WHAT SHORTCUTS DO YOU USE ON A REGULAR BASIS?
WHAT IS THE LINUX STANDARD BASE?
LINUX STANDARD BASE (LSB) IS A JOINT PROJECT BY SEVERAL LINUX DISTRIBUTIONS UNDER THE ORGANIZATIONAL STRUCTURE OF THE LINUX FOUNDATION TO STANDARDIZE THE SOFTWARE SYSTEM STRUCTURE, INCLUDING THE FILESYSTEM HIERARCHY STANDARD USED IN THE LINUX KERNEL.

WHAT IS AN ATOMIC OPERATION?
ATOMIC OPERATIONS IN CONCURRENT PROGRAMMING ARE PROGRAM OPERATIONS THAT RUN COMPLETELY INDEPENDENTLY OF ANY OTHER PROCESSES. ATOMIC OPERATIONS ARE USED IN MANY MODERN OPERATING SYSTEMS AND PARALLEL PROCESSING SYSTEMS.

YOUR FRESHLY CONFIGURED HTTP SERVER IS NOT RUNNING AFTER A RESTART, WHAT CAN YOU DO?
TROUBLESHOOT. CHECK IF THE SERVICE IS RUNNING. CHECK FOR MISCONFIGURATIONS. CHECK THE LOGS. ETC. CHECK MY OWN CONNECTION.

WHAT KIND OF KEYS ARE IN ~/.SSH/AUTHORIZED_KEYS AND WHAT IT IS THIS FILE USED FOR?
AUTHORIZED_KEYS FILE IN SSH. THE AUTHORIZED_KEYS FILE IN SSH SPECIFIES THE SSH KEYS THAT CAN BE USED FOR LOGGING INTO THE USER ACCOUNT FOR WHICH THE FILE IS CONFIGURED. IT IS A HIGHLY IMPORTANT CONFIGURATION FILE, AS IT CONFIGURES PERMANENT ACCESS USING SSH KEYS AND NEEDS PROPER MANAGEMENT.

I'VE ADDED MY PUBLIC SSH KEY INTO AUTHORIZED_KEYS BUT I'M STILL GETTING A PASSWORD PROMPT, WHAT CAN BE WRONG?
MAKE SURE THE PERMISSIONS ON THE ~/.SSH DIRECTORY AND ITS CONTENTS ARE PROPER. WHEN I FIRST SET UP MY SSH KEY AUTH, I DIDN'T HAVE THE ~/.SSH FOLDER PROPERLY SET UP, AND IT YELLED AT ME.

YOUR HOME DIRECTORY ~, YOUR ~/.SSH DIRECTORY AND THE ~/.SSH/AUTHORIZED_KEYS FILE ON THE REMOTE MACHINE MUST BE WRITABLE ONLY BY YOU: RWX------ AND RWXR-XR-X ARE FINE, BUT RWXRWX--- IS NO GOOD[1], EVEN IF YOU ARE THE ONLY USER IN YOUR GROUP (IF YOU PREFER NUMERIC MODES: 700 OR 755, NOT 775). IF ~/.SSH OR AUTHORIZED_KEYS IS A SYMBOLIC LINK, THE CANONICAL PATH (WITH SYMBOLIC LINKS EXPANDED) IS CHECKED. YOUR ~/.SSH/AUTHORIZED_KEYS FILE (ON THE REMOTE MACHINE) MUST BE READABLE (AT LEAST 400), BUT YOU'LL NEED IT TO BE ALSO WRITABLE (600) IF YOU WILL ADD ANY MORE KEYS TO IT. YOUR PRIVATE KEY FILE (ON THE LOCAL MACHINE) MUST BE READABLE AND WRITABLE ONLY BY YOU: RW-------, I.E. 600. ALSO, IF SELINUX IS SET TO ENFORCING, YOU MAY NEED TO RUN RESTORECON -R -V ~/.SSH (SEE E.G. UBUNTU BUG 965663 AND DEBIAN BUG REPORT #658675; THIS IS PATCHED IN CENTOS 6).

DID YOU EVER CREATE RPM'S, DEB'S OR SOLARIS PKG'S?

WHAT DOES :(){ :|:& };: DO ON YOUR SYSTEM?

THIS IS CALLED A FORK BOMB.

:()MEANS YOU ARE DEFINING A FUNCTION CALLED:

{:|: &} MEANS RUN THE FUNCTION : AND SEND ITS OUTPUT TO THE : FUNCTION AGAIN AND RUN THAT IN THE BACKGROUND.

THE ; IS A COMMAND SEPARATOR, LIKE &&.

: RUNS THE FUNCTION THE FIRST TIME.

ESSENTIALLY YOU ARE CREATING A FUNCTION THAT CALLS ITSELF TWICE EVERY CALL AND DOESN'T HAVE ANY WAY TO TERMINATE ITSELF. IT WILL KEEP DOUBLING UP UNTIL YOU RUN OUT OF SYSTEM RESOURCES.

HOW DO YOU CATCH A LINUX SIGNAL ON A SCRIPT?

```
TRAP SIGUSR1 USR1       # CATCH -USR1 SIGNAL
```

CAN YOU CATCH A SIGKILL?

YOU CAN'T CATCH SIGKILL (AND SIGSTOP), SO ENABLING YOUR CUSTOM HANDLER FOR SIGKILL IS MOOT.

WHAT'S HAPPENING WHEN THE LINUX KERNEL IS STARTING THE OOM KILLER AND HOW DOES IT CHOOSE WHICH PROCESS TO KILL FIRST?

DESCRIBE THE LINUX BOOT PROCESS WITH AS MUCH DETAIL AS POSSIBLE, STARTING FROM WHEN THE SYSTEM IS POWERED ON AND ENDING WHEN YOU GET A PROMPT.

WHAT'S A CHROOT JAIL?

A CHROOT JAIL IS A WAY TO ISOLATE A PROCESS AND ITS CHILDREN FROM THE REST OF THE SYSTEM. IT SHOULD ONLY BE USED FOR PROCESSES THAT DON'T RUN AS ROOT, AS ROOT USERS CAN BREAK OUT OF THE JAIL VERY EASILY.

WHEN TRYING TO UMOUNT A DIRECTORY IT SAYS IT'S BUSY, HOW TO FIND OUT WHICH PID HOLDS THE DIRECTORY?

OPEN A TERMINAL:

```
FUSER -C /MEDIA/KINGSTON
```

IT WILL OUTPUT SOMETHING LIKE THIS:

```
/MEDIA/KINGSTON/: 3106C 11086
```

THIS WILL GIVE YOU THE PID OF THE PROCESSES USING THIS VOLUME. THE EXTRA CHARACTER AT THE END OF PID WILL GIVE SOME EXTRA INFO. (C IN 3106C)

C - THE PROCESS IS USING THE FILE AS ITS CURRENT WORKING DIRECTORY M - THE FILE IS MAPPED WITH MMAP O - THE PROCESS IS USING IT AS AN OPEN FILE R - THE FILE IS THE ROOT DIRECTORY OF THE PROCESS T - THE PROCESS IS ACCESSING THE FILE AS A TEXT FILE Y - THIS FILE IS THE CONTROLLING TERMINAL FOR THE PROCESS

SO TO UNMOUNT JUST KILL THAT PIDS AND RE-TRY THE UNMOUNT

```
SUDO KILL -9 3106 11086 SUDO UMOUNT /MEDIA/KINGSTON
```

WHAT'S LD_PRELOAD AND WHEN IT'S USED?

IF YOU SET LD_PRELOAD TO THE PATH OF A SHARED OBJECT, THAT FILE WILL BE LOADED BEFORE ANY OTHER LIBRARY (INCLUDING THE C RUNTIME, LIBC.SO). SO TO RUN LS WITH YOUR SPECIAL MALLOC() IMPLEMENTATION, DO THIS:

```
$ LD_PRELOAD=/PATH/TO/MY/MALLOC.SO /BIN/LS
```

YOU RAN A BINARY AND NOTHING HAPPENED. HOW WOULD YOU DEBUG THIS?
GDB DEBUGGER OR CHECK THE RETURN CODE ECHO $?

WHAT ARE CGROUPS? CAN YOU SPECIFY A SCENARIO WHERE YOU COULD USE THEM?
CGROUPS (ABBREVIATED FROM CONTROL GROUPS) IS A LINUX KERNEL FEATURE THAT LIMITS, ACCOUNTS FOR, AND ISOLATES THE RESOURCE USAGE (CPU, MEMORY, DISK I/O, NETWORK, ETC.) OF A COLLECTION OF PROCESSES.

HOW CAN YOU REMOVE/DELETE A FILE WITH FILE-NAME CONSISTING OF ONLY NON-PRINTABLE/NON-TYPE-ABLE CHARACTERS?
THE FILE HAS A NAME, BUT IT'S MADE OF NON-PRINTABLE CHARACTERS. IF YOU USE KSH93, BASH, ZSH, MKSH OR FREEBSD SH, YOU CAN TRY TO REMOVE IT BY SPECIFYING ITS NON-PRINTABLE NAME. FIRST ENSURE THAT THE NAME IS RIGHT WITH: LS -LD
'\177'

ANOTHER (A BIT MORE RISKY) APPROACH IS TO USE RM -I -- * . WITH THE -I OPTION RM REQUIRES CONFIRMATION BEFORE REMOVING A FILE, SO YOU CAN SKIP ALL FILES YOU WANT TO KEEP BUT THE ONE.

HOW CAN YOU INCREASE OR DECREASE THE PRIORITY OF A PROCESS IN LINUX?

```
NICE -N 10 APT-GET UPGRADE
```

WHAT ARE RUN-LEVELS IN LINUX?
A RUNLEVEL IS ONE OF THE MODES THAT A UNIX -BASED OPERATING SYSTEM WILL RUN IN. EACH RUNLEVEL HAS A CERTAIN NUMBER OF SERVICES STOPPED OR STARTED, GIVING THE USER CONTROL OVER THE BEHAVIOR OF THE MACHINE. CONVENTIONALLY, SEVEN RUNLEVELS EXIST, NUMBERED FROM ZERO TO SIX.

AFTER THE LINUX KERNEL HAS BOOTED, THE INIT PROGRAM READS THE /ETC/INITTAB FILE TO DETERMINE THE BEHAVIOR FOR EACH RUNLEVEL. UNLESS THE USER SPECIFIES ANOTHER VALUE AS A KERNEL BOOT PARAMETER, THE SYSTEM WILL ATTEMPT TO ENTER (START) THE DEFAULT RUNLEVEL.

RUN LEVEL	MODE	ACTION
0	HALT	SHUTS DOWN SYSTEM
1	SINGLE-USER MODE	DOES NOT CONFIGURE NETWORK INTERFACES, START DAEMONS, OR ALLOW NON-ROOT LOGINS
2	MULTI-USER MODE	DOES NOT CONFIGURE NETWORK INTERFACES OR START DAEMONS.
3	MULTI-USER MODE WITH NETWORKING	STARTS THE SYSTEM NORMALLY.
4	UNDEFINED	NOT USED/USER-DEFINABLE
5	X11	AS RUNLEVEL 3 + DISPLAY MANAGER(X)
6	REBOOT	REBOOTS THE SYSTEM

ADMIN-LEVEL:
EXPLAIN THE LOGICAL STEPS TO INCREASE THE SIZE OF LVM PARTITION?
ANSWER: SOME LOGICAL STEPS NEED TO BE FOLLOWED TO INCREASE THE SIZE OF LVM PARTITION. THESE ARE AS FOLLOWS:

RUN THE COMMAND AS PER GIVEN FORMAT:

```
LVEXTEND -L +500M /DEV/<NAME OF THE LVM PARTITION>
```

HERE, WE ARE EXTENDING THE SIZE OF LVM PARTITION BY 500MB.

```
RESIZE2FS /DEV/
```

YOU CAN CHECK THE SIZE OF PARTITION USING ‘DF -H’ COMMAND
USING WHICH UTILITY YOU CAN CREATE A PARTITION FROM THE RAW DISK?
ANSWER: TO CREATE THE PARTITION FROM THE RAW DISK, YOU HAVE TO USE FDISK UTILITY.BELOW ARE THE STEPS TO CREATE A PARTITION FROM THE RAW DISK:

STEP 1: RUN THE BELOW COMMAND:

```
FDISK  /DEV/HD* (IDE) OR /DEV/SD* (SCSI)
```

STEP 2: TYPE N TO CREATE A NEW PARTITION

STEP 3: NOW PARTITION HAS BEEN CREATED, AND WE HAVE TO WRITE THE CHANGES TO THE PARTITION TABLE, SO TYPE W COMMAND TO WRITE THE CHANGES.

HOW DO YOU CREATE A NEW USER ACCOUNT AND SET THE PASSWORD FOR A USER FROM A SHELL PROMPT IN LINUX?

ANSWER: TO CREATE A NEW USER ACCOUNT FROM A SHELL PROMPT FOLLOW THE BELOW STEPS:

LOG IN AS ROOT USER IF YOU ARE NOT LOGGED IN AS ROOT USING SU – COMMAND.

ENTER THE ROOT USER PASSWORD

THE USERADD COMMAND IS USED TO CREATE A NEW USER IN LINUX. SO, TYPE COMMAND USERADD AND GIVE THE USERNAME YOU WANT TO CREATE AS GIVEN BELOW:

```
USERADD SMITH
```

TO SET THE PASSWORD OF THE USER SMITH TYPE THE COMMAND: PASSWD SMITH

IT WILL PROMPT FOR THE NEW PASSWORD. ENTER THE NEW PASSWORD FOR USER SMITH.

IT WILL ASK TO RETYPE THE PASSWORD. SO, RETYPE THE SAME PASSWORD AND PASSWORD IS SET FOR THE USER.

WHAT ARE THE DEFAULT PORT NUMBERS USED FOR SMTP, FTP,DNS, DHCP, SSH?

ANSWER:

SERVICE PORT SMTP 25 FTP 20 FOR DATA TRANSFER AND 21 FOR CONNECTION ESTABLISHED

DNS 53 DHCP 67/UDP(FOR DHCP SERVER, 68/UDPFOR DHCP CLIENT SSH 22

EXPLAIN ALL THE FIELDS IN THE/ETC/PASSWD FILE?

ANSWER: /ETC/PASSWD FILE CONTAINS THE USEFUL INFORMATION FOR ALL THE SYSTEM USERS WHO LOG IN. WE HAVE MANY FIELDS IN /ETC/PASSWD FILE SUCH AS USERNAME, PASSWORD, USER ID, GROUP ID, COMMENT OR USER ID INFO, HOME DIRECTORY, COMMAND /SHELL, ETC. SO, THIS FILE CONTAINS SENSITIVE INFORMATION REGARDING ALL THE USER ACCOUNTS. THERE IS A SINGLE LINE PER USER IN THIS FILE. COLON (:) SEPARATES THE FIELDS IN /ETC/PASSWD. BELOW IS THE EXPLANATION OF THE FIELDS.

USERNAME: FIRST FIELD IS THE USERNAME THAT CONTAINS THE USERNAME WHICH IS 1 TO 32 LENGTH CHARACTERS.

PASSWORD: THIS FIELD DOES NOT SHOW THE ACTUAL PASSWORD AS THE PASSWORD IS ENCRYPTED. HERE, X CHARACTER SHOWS THAT PASSWORD IS ENCRYPTED THAT IS LOCATED IN /ETC/SHADOW FILE.

USER ID (UID): ALL THE USERS CREATED IN LINUX IS GIVEN A USER ID WHENEVER THE USER IS CREATED. UID 0 IS FIXED AND RESERVED FOR THE ROOT USER.

GROUP ID (GID): THIS FIELD SPECIFIES THE NAME OF THE GROUP TO WHICH THE USER BELONGS. THE GROUP INFORMATION IS ALSO STORED IN A FILE /ETC/GROUP.

USER ID INFO: HERE YOU CAN ADD COMMENTS AND YOU CAN ADD ANY EXTRA INFORMATION RELATED TO THE USERS LIKE FULL NAME, CONTACT NUMBER, ETC.

HOME DIRECTORY: THIS FIELD PROVIDES THE PATH WHERE THE USER IS DIRECTED AFTER THE LOGIN. FOR EXAMPLE, /HOME/SMITH.

COMMAND/SHELL: THIS FIELD PROVIDES THE PATH OF A COMMAND/SHELL AND DENOTES THAT USER HAS ACCESS TO THIS SHELL I.E. /BIN/BASH.

HOW CAN AN ADMINISTRATOR KNOW WHETHER A USER ACCOUNT IS LOCKED OR NOT?

ANSWER: TO CHECK IF THE USER ACCOUNT IS LOCKED OR NOT JUST RUN THIS COMMAND IN THE SHELL:

```
PASSWD –S
```

OR SEARCH FOR THE GREP USERNAME IN THE LOCATION /ETC/SHADOW FILE AND IT WILL SHOW A SYMBOL ‘!’ PREFIX TO THE ENCRYPTED FIELD IN THE PASSWORD BOX.

TO JUST UNLOCK THE PASSWORD TYPE THIS COMMAND:

```
PASSWD –U
```

IF THERE IS A DOUBLE EXCLAMATION MARK THEN RUN THIS COMMAND TWO TIMES:

```
USERMOD –U
```

WHAT DO YOU MEAN BY SELINUX?
ANSWER: SELINUX IS THE ABBREVIATION FOR SECURITY ENHANCED LINUX. THE ACCESS CONTROLS FOR THE USERS CAN BE CONTROLLED USING SELINUX. FOR EXAMPLE, THE USERS CAN BE STOPPED FROM RUNNING THE SCRIPTS AND ACCESSING THEIR OWN HOME DIRECTORIES. SELINUX HAS THE CAPABILITY TO SUPPORT THE ACCESS CONTROL AND SECURITY POLICIES. IT BASICALLY OPERATES ON THREE DIFFERENT MODES:

ENFORCING –TO ENFORCE ITS POLICIES.
PERMISSIVE –POLICES WANT TO APPLY BUT WILL BE LOCKED IN CASE OF VIOLATION.
DISABLED –SELINUX WILL STAY IN DISABLED MODE.
TO CHECK THE STATUS OF SELINUX, JUST TYPE: #GETENFORE OR SESTATUS

MENTION THE RUN LEVELS IN LINUX AND STEPS TO EDIT THEM.
ANSWER: RUN LEVELS ARE IDENTIFIED BY NUMBERS IN LINUX. THE RUN LEVELS DETERMINE WHAT ARE THE SERVICES THAT ARE CURRENTLY IN OPERATION. THERE ARE SEVEN DIFFERENT RUN LEVELS IN LINUX:

HALT SYSTEM
SINGLE USER MODE
USER MULTI-MODE EXCLUDING NFS
FULL MULTI-USER MODE
UNUSED
MULTI-USER MODE (GRAPHICAL USER MODE)
REBOOT SYSTEM
TO CHANGE THE EDIT LEVEL /ETC/INITTLAB AND EDIT THE INITDEFAULT ENTRY.

HOW CAN WE CREATE A LOCAL YUM REPOSITORY IN THE LOCATION /MEDIA WITH THE USE OF MOUNTED LINUX ISO IMAGE?
ANSWER: TO CREATE THE LOCAL YUM REPOSITORY YOU HAVE TO CREATE THE FILES ENDING WITH EXTENSION .REPO IN THE LOCATION /ETC/YUM.REPOS.D

```
SYNTAX: [ROOT@LOCALHOST YUM.REPOS.D]# CAT LOCAL.REPO

[LOCAL]

NAME=RHEL6.5

BASEURL=FILE:///MEDIA

ENABLED=1

GPGCHECK=1

GPGKEY=FILE:///MEDIA/RPM-GPG-KEY-REDHAT-RELEASE
```

MENTION THE METHODS TO CHECK WHETHER USING YUM, THE PACKAGE IS INSTALLED SUCCESSFULLY OR NOT.
ANSWER: THERE ARE SEVERAL METHODS TO CHECK WHETHER THE PACKAGE IS INSTALLED OR NOT. TO UNDERSTAND, JUST SEE THE BELOW STEPS.

METHOD 1 –IF THE COMMAND IS EXECUTED SUCCESSFULLY THEN AFTER RUNNING THE YUM

COMMAND IT WILL SHOW ‘0’ ON CHECKING THE EXIT STATUS.

METHOD 2-YOU CAN RUN THE RPM AND –QA TEST.

METHOD 3–IN THE YUM LOG, CHECK IF ANY ENTRY IS INSTALLED IN THE DIRECTORY.

WHAT IS THE DIFFERENCE BETWEEN HARD LINK AND SOFT LINK?
ANSWER: A SOFT LINK(SYMBOLIC LINK) POINTS TO ANOTHER FILE BY NAME. AS IT JUST CONTAINS A NAME, THAT NAME DOES NOT ACTUALLY HAVE TO EXIST OR EXIST ON A DIFFERENT FILE SYSTEM. IF YOU REPLACE THE FILE OR CHANGE FILE CONTENT WITHOUT CHANGING A NAME, THEN THE LINK STILL CONTAINS THE SAME NAME AND POINTS TO THAT FILE. A HARD LINK POINTS TO THE FILE BY INODE NUMBER. A FILE SHOULD ACTUALLY EXIST IN THE SAME FILE SYSTEM. A FILE WILL ONLY BE DELETED FROM DISK WHEN THE LAST LINK TO ITS INODE IS REMOVED.

A RUNNING PROCESS GETS EAGAIN: RESOURCE TEMPORARILY UNAVAILABLE ON READING A SOCKET. HOW CAN YOU CLOSE THIS BAD SOCKET/FILE DESCRIPTOR WITHOUT KILLING THE PROCESS?
GET THE FILE DESCRIPTOR OF THE SOCKET, DEBUG THE PROCESS AND MANUALLY CALL CLOSE ON THE FILE DESCRIPTOR.

ON LINUX SYSTEMS:

FIND THE OFFENDING PROCESS: NETSTAT -NP
FIND THE SOCKET FILE DESCRIPTOR: LSOF -NP $PID
DEBUG THE PROCESS: GDB -P $PID
CLOSE THE SOCKET: CALL CLOSE($FD)
CLOSE THE DEBUGGER: QUIT
PROFIT.
FROM HERE.

WHAT DO YOU CONTROL WITH SWAPINESS?
THE LINUX KERNEL PROVIDES A TWEAKABLE SETTING THAT CONTROLS HOW OFTEN THE SWAP FILE IS USED, CALLED SWAPPINESS.

A SWAPPINESS SETTING OF ZERO MEANS THAT THE DISK WILL BE AVOIDED UNLESS ABSOLUTELY NECESSARY (YOU RUN OUT OF MEMORY), WHILE A SWAPPINESS SETTING OF 100 MEANS THAT PROGRAMS WILL BE SWAPPED TO DISK ALMOST INSTANTLY.

HOW DO YOU CHANGE TCP STACK BUFFERS? HOW DO YOU CALCULATE IT?
TCP TUNING HTTP://WWW.LINUX-ADMINS.NET/2010/09/LINUX-TCP-TUNING.HTML

WHAT IS HUGE TABLES? WHY ISN'T IT ENABLED BY DEFAULT? WHY AND WHEN USE IT?
HUGEPAGES FEATURE ENABLES THE LINUX KERNEL TO MANAGE LARGE PAGES OF MEMORY IN ADDITION TO THE STANDARD 4KB (ON X86 AND X86_64) OR 16KB (ON IA64) PAGE SIZE. IF YOU HAVE A SYSTEM WITH MORE THAN 16GB OF MEMORY RUNNING ORACLE DATABASES WITH A TOTAL SYSTEM GLOBAL AREA (SGA) LARGER THAN 8GB, YOU SHOULD ENABLE THE HUGEPAGES FEATURE TO IMPROVE DATABASE PERFORMANCE.

WHAT IS LUKS? HOW TO USE IT?
LUKS IS THE STANDARD FOR LINUX HARD DISK ENCRYPTION. BY PROVIDING A STANDARD ON-DISK-FORMAT, IT DOES NOT ONLY FACILITATE COMPATIBILITY AMONG DISTRIBUTIONS, BUT ALSO PROVIDES SECURE MANAGEMENT OF MULTIPLE USER PASSWORDS.

TECHNICAL:
WHAT IS THE ADVANTAGE OF EXECUTING THE RUNNING PROCESSES IN THE BACKGROUND? HOW CAN YOU DO THAT?
ANSWER: THE MOST SIGNIFICANT ADVANTAGE OF EXECUTING THE RUNNING PROCESS IN THE BACKGROUND IS THAT YOU CAN DO ANY OTHER TASK SIMULTANEOUSLY WHILE OTHER PROCESSES

ARE RUNNING IN THE BACKGROUND. SO, MORE PROCESSES CAN BE COMPLETED IN THE BACKGROUND WHILE YOU ARE WORKING ON DIFFERENT PROCESSES. IT CAN BE ACHIEVED BY ADDING A SPECIAL CHARACTER ‘&’ AT THE END OF THE COMMAND.

LIST THE DIFFERENCES BETWEEN BASH AND DOS?
ANSWER: THERE ARE MANY DIFFERENCES BETWEEN BASH AND DOS THAT ARE AS BELOW:

OUT OF THESE TWO COMMANDS, BASH IS CASE SENSITIVE WHILE DOS IS NOT CASE SENSITIVE.
IN BASH ‘/’ ACTS THE DIRECTORY SEPARATOR WHILE IN DOS ‘/’ ACTS AS THE COMMAND ARGUMENT DELIMITER.
IN BASH ‘\’ IS USED AS THE ESCAPE CHARACTER WHILE IN DOS ‘\’ ACTS AS THE DIRECTORY SEPARATOR.
IN BASH, THERE IS A FILE CONVENTION USED WHILE IN DOS, THERE IS NO ANY FILE CONVENTION USED.
HOW CAN MULTIPLE MACHINES SHARE A SINGLE INTERNET CONNECTION IN LINUX?
ANSWER: LINUX MACHINE CAN BE MADE AS A ROUTER SO THAT MULTIPLE DEVICES CAN SHARE A SINGLE INTERNET CONNECTION. FOR THIS, WE HAVE TO USE A FEATURE CALLED “IP MASQUERADE.” THIS FUNCTIONALITY WILL HELP TO CONNECT MULTIPLE COMPUTERS TO CONNECT TO THE LINUX MACHINE AS WELL AS INTERNET. THIS FUNCTIONALITY WILL ALSO ALLOW THOSE INTERNAL COMPUTERS TO CONNECT WHO DO NOT HAVE IP ADDRESSES.

IF A VOLUME GROUP ALREADY EXISTS AND WE NEED TO EXTEND THE VOLUME GROUP TO SOME EXTENT. HOW WILL YOU ACHIEVE THIS?
ANSWER: LINUX PROVIDE THE FACILITY TO INCREASE THE SIZE OF A VOLUME GROUP EVEN IF IT ALREADY EXISTS. FOR THIS, WE NEED TO RUN A COMMAND.

FIRST OF ALL, WE HAVE TO CREATE A PHYSICAL VOLUME (/DEV/SDA1)

SIZE OF THE PHYSICAL VOLUME SHOULD BE THE SIZE YOU WANT THE SIZE OF THE LOGICAL VOLUME.

NOW, RUN THE BELOW COMMAND:

```
VGEXTEND VG1 /DEV/SDA1
```

HERE VG1 IS THE NAME OF THE VOLUME GROUP.

WHY IS “FINGER SERVICE” ALWAYS KEPT DISABLED WHEN NOT IN USE?
ANSWER: FINGER SERVICE ACTS AS BOTH THE WEB AND FTP SERVER. IT IS ALSO KNOWN AS FINGER USER INFORMATION PROTOCOL WHICH CONTAINS THE INFORMATION OF THE USER THAT CAN BE VIEWED BY THE CLIENTS. IT ALLOWS A REMOTE USER TO SEE THE INFORMATION ABOUT THE ADMIN SUCH AS LOGIN SHELL, LOGIN NAME AND OTHER CONFIDENTIAL DETAILS. THAT IS WHY, THE FINGER SERVICE SHOULD BE KEPT DISABLED WHEN IT’S NOT IN USE.

IF IT IS NOT DISABLED, YOU HAVE TO MODIFY AND COMMENT OUT THE FILE “/ETC/INETD.CONF”.

HOW CAN WE MAKE A ROUTER WITH THE HELP OF LINUX COMPUTER?
ANSWER: YOU MAY GENERALLY COME ACROSS THIS TYPE OF QUESTIONS IN LINUX INTERVIEW. LINUX MACHINE HAS THE ABILITY TO TURN IT INTO A ROUTER WITH THE HELP OF IP MASQUERADE. YOU MAY HAVE SEEN THE SERVERS FOUND IN COMMERCIAL FIREWALLS. IP MASQUERADE DOES THE SAME FUNCTION TO ONE-TO-MANY NETWORK ADDRESS TRANSLATION SERVERS. IF THE INTERNAL COMPUTERS DO NOT HAVE THE IP ADDRESS THEN IN THIS CASE, IP MASQUERADE CAN CONNECT TO THE OTHER INTERNAL COMPUTERS WHICH ARE CONNECTED TO LINUX BOX TO ACCESS THE INTERNET.

JUST FOLLOW THESE STEPS TO ENABLE IP MASQUERADE LINUX:

CONNECT YOUR PC TO LAN.
THIS PC CAN BE USED AS A DEFAULT GATEWAY FOR OTHER SYSTEMS FOR TCP/IP NETWORKING.

YOU CAN USE THE SAME DNS ON ALL OTHER SYSTEMS.
GO IN THE KERNEL AND ENABLE IP FORWARDING. YOU CAN ALSO ENABLE IP FORWARDING USING THE COMMAND: /ETC/RC.D/RC.LOCAL FILE ON REBOOTING THE SYSTEM.
THE LAST STEP IS TO RUN THIS COMMAND WHICH SETS UP THE RULES TO MASQUERADE: /SBIN/IPTABLES
HOW WE CAN ENABLE ACL?
ANSWER: ACL IS AN ACRONYM FOR ACCESS CONTROL LIST WHICH IS USED TO PROVIDE FLEXIBLE PERMISSION MECHANISM FOR THE FILE SYSTEMS. WE CAN ENABLE ACL BY FOLLOWING METHODS:

TYPE THE CODE IN THE SHELL: /ETC/FSTAB WITH A LABEL=/HOME/EXT3 ACL

NOW WE HAVE TO REMOUNT THIS FILE SYSTEM WITH THE ACL PARTITION: MOUNT –T EXT3 –O ACL /DEV/SDA3/HOME

WHAT DO YOU MEAN BY REDIRECTION?
ANSWER: WHEN THE DATA IS DIRECTED FROM ONE OUTPUT TO ANOTHER OUTPUT EVEN WHEN THE OUTPUT WILL SERVE THE DATA AS AN INPUT FOR ANOTHER PROCESS, THIS IS CALLED REDIRECTION.

WHAT IS COMMAND GROUPING?
ANSWER: WE CAN REDIRECT A COMMAND FROM A FILE OR TO A FILE. IT IS USUALLY DONE WITH THE HELP OF BRACES OR PARENTHESIS. WHEN THE COMMAND IS GROUPED THEN REDIRECTION IS DONE TO THE WHOLE GROUP.

THE COMMAND IS EXECUTED BY THE CURRENT SHELL WHEN WE USE THE BRACES () AND IN CASE WE HAVE TO EXECUTE A COMMAND BY A SUBSHELL THEN WE USE PARENTHESIS {}.

EXPLAIN FILE PERMISSION IN LINUX. HOW TO CHANGE IT?
ANSWER: PERMISSIONS ARE ESTABLISHED FOR ALL FILES AND DIRECTORIES. PERMISSIONS SPECIFY WHO CAN ACCESS A FILE OR DIRECTORY, AND THE TYPES OF ACCESS. ALL FILES AND DIRECTORIES ARE OWNED BY A USER.

PERMISSIONS ARE CONTROLLED AT THREE LEVELS:

OWNER (CALLED A USER, OR ‘U’)
GROUP (‘G’)
THE REST USERS(CALLED OTHER, OR ‘O’)
LEVEL OF ACCESS:

READ – FILET CAN BE VIEWED OR COPIED.
WRITE – FILE CAN BE OVERWRITTEN (E.G., USING SAVE AS)
EXECUTE – FILE CAN BE EXECUTED
TO CHANGE PERMISSION – CHMOD < FILE(S)> IS USED. HERE PERMISSIONS CAN BE SPECIFIED DIFFERENT APPROACHES. THE PARAMETER FILE(S) IS ONE OR MORE FILES (OR DIRECTORIES). ONE APPROACH TO SPECIFY PERMISSIONS IS TO DESCRIBE THE CHANGES TO BE APPLIED AS A COMBINATION OF U, G, O ALONG WITH R, W, X. TO ADD PERMISSION, USE + AND TO REMOVE PERMISSION, USE –.

WHAT IS THE PROCESS IN A LINUX CONTEXT?
ANSWER: A PROCESS IS A RUNNING PROGRAM. PROCESSES CAN BE STARTED FROM THE GUI OR THE COMMAND LINE. PROCESSES CAN ALSO START OTHER PROCESSES. WHENEVER A PROCESS RUNS, LINUX KEEPS TRACK OF IT THROUGH A PROCESS ID (PID). AFTER BOOTING, THE FIRST PROCESS IS AN INITIALIZATION PROCESS CALLED INIT. IT IS GIVEN A PID OF 1. FROM THAT POINT ON, EACH NEW PROCESS GETS THE NEXT AVAILABLE PID.

A PROCESS CAN ONLY BE CREATED BY ANOTHER PROCESS. WE REFER TO THE CREATING PROCESS AS THE PARENT AND THE CREATED PROCESS AS THE CHILD. THE PARENT PROCESS SPAWNS ONE OR MORE CHILD PROCESSES. THE SPAWNING OF A PROCESS CAN BE ACCOMPLISHED IN ONE OF SEVERAL WAYS. EACH REQUIRES A SYSTEM CALL (FUNCTION CALL) TO THE LINUX KERNEL.

THESE FUNCTION CALLS ARE FORK(), VFORK(), CLONE(), WAIT(), AND EXEC().

KERNEL:
WHAT DO YOU UNDERSTAND ABOUT LINUX KERNEL AND CAN YOU EDIT IT?
ANSWER: LINUX KERNEL IS THE COMPONENT THAT MANAGES THE HARDWARE RESOURCES FOR THE USER AND THAT PROVIDES ESSENTIAL SERVICES AND INTERACT WITH THE USER COMMANDS. LINUX KERNEL IS AN OPEN SOURCE SOFTWARE AND FREE, AND IT IS RELEASED UNDER GENERAL PUBLIC LICENSE SO WE CAN EDIT IT AND IT IS LEGAL.

WHAT ARE THE DIFFERENT TYPES OF KERNELS? EXPLAIN
ANSWER: WE CAN BUILD KERNELS BY MANY DIFFERENT TYPES, BUT 3 OF THE TYPES OF KERNELS ARE MOST COMMONLY USED: MONOLITHIC, MICROKERNEL AND HYBRID.

MICROKERNEL: THIS TYPE OF KERNEL ONLY MANAGES CPU, MEMORY, AND IPC. THIS KIND OF KERNEL PROVIDES PORTABILITY, SMALL MEMORY FOOTPRINT AND ALSO SECURITY.

MONOLITHIC KERNEL: LINUX IS A MONOLITHIC KERNEL. SO, THIS TYPE OF KERNEL PROVIDES FILE MANAGEMENT, SYSTEM SERVER CALLS, ALSO MANAGES CPU, IPC AS WELL AS DEVICE DRIVERS. IT PROVIDES EASIER ACCESS TO THE PROCESS TO COMMUNICATE AND AS THERE IS NOT ANY QUEUE FOR PROCESSOR TIME, SO PROCESSES REACT FASTER.

HYBRID KERNEL: IN THIS TYPE OF KERNELS, PROGRAMMERS CAN SELECT WHAT THEY WANT TO RUN IN USER MODE AND WHAT IN SUPERVISOR MODE. SO, THIS KERNEL PROVIDES MORE FLEXIBILITY THAN ANY OTHER KERNEL BUT IT CAN HAVE SOME LATENCY PROBLEMS.

EXPLAIN LINUX BOOT SEQUENCE.
ANSWER: THERE ARE SIX LEVELS OF A LINUX BOOT SEQUENCE. THESE ARE AS FOLLOWS:

BIOS: FULL FORM OF BIOS IS BASIC INPUT OR OUTPUT SYSTEM THAT PERFORMS INTEGRITY CHECKS AND IT WILL SEARCH AND LOAD AND THEN IT WILL EXECUTE THE BOOTLOADER.

MBR: MBR MEANS MASTER BOOT RECORD. MBR CONTAINS THE INFORMATION REGARDING GRUB AND EXECUTES AND LOADS THIS BOOTLOADER.

GRUB: GRUB MEANS GRAND UNIFIED BOOTLOADER. IN CASE, MANY KERNEL IMAGES ARE INSTALLED ON YOUR SYSTEM THEN YOU CAN SELECT WHICH ONE YOU WANT TO EXECUTE.

KERNEL: ROOT FILE SYSTEM IS MOUNTED BY KERNEL AND EXECUTES THE /SBIN/INIT PROGRAM.

INIT: INIT CHECKS THE FILE /ETC/INITTAB AND DECIDES THE RUN LEVEL. THERE ARE SEVEN-RUN LEVELS AVAILABLE FROM 0-6. IT WILL IDENTIFY THE DEFAULT INIT LEVEL AND WILL LOAD THE PROGRAM.

RUNLEVEL PROGRAMS: AS PER YOUR DEFAULT SETTINGS FOR THE RUN LEVEL, THE SYSTEM WILL EXECUTE THE PROGRAMS.

EXPLAIN INTERRUPTS IN LINUX AND ALSO EXPLAIN INTERRUPT HANDLERS.
ANSWER: INTERRUPTS MEANS THE PROCESSOR IS TRANSFERRED TEMPORARILY TO ANOTHER PROGRAM OR FUNCTION. WHEN THAT PROGRAM IS COMPLETED, THE PROCESSOR WILL BE GIVEN BACK TO THAT PROGRAM TO COMPLETE THE TASK.

INTERRUPT HANDLER IS THE FUNCTION THAT THE KERNEL RUNS FOR A SPECIFIC INTERRUPT. IT IS ALSO CALLED INTERRUPT SERVICE ROUTINE. INTERRUPTS HANDLERS ARE THE FUNCTION THAT MATCHES A PARTICULAR PROTOTYPE AND ENABLES THE KERNEL TO PASS THE HANDLER INFORMATION ACCURATELY.

WHAT IS PAGE FRAME?
ANSWER: A PAGE FRAME IS A BLOCK OF RAM THAT IS USED FOR VIRTUAL MEMORY. IT HAS ITS PAGE FRAME NUMBER. THE SIZE OF A PAGE FRAME MAY VARY FROM SYSTEM TO SYSTEM, AND IT

IS IN THE POWER OF 2 IN BYTES. ALSO, IT IS THE SMALLEST LENGTH BLOCK OF MEMORY IN WHICH AN OPERATING SYSTEM MAPS MEMORY PAGES.

WHAT ARE THE POSSIBLE METHODS TO DEPLOY A MODULE INSIDE A KERNEL?
ANSWER: TO CHECK THE MODULES THAT ARE ALREADY INSTALLED INSIDE THE KERNEL, YOU HAVE TO RUN THIS CODE: LSMOD. WHEN THE MODULE HAS BEEN BUILT, NOW IT IS THE STAGE TO LOAD IT IN THE KERNEL. YOU CAN LOAD IT BY THE COMMAND “INSMOD” OR “MODPROBE”.

SYNTAX: INSMOD[FILENAME][MODULE-OPTIONS] //MODULE-OPTIONS ARE COMMAND LINE ARGUMENTS TO KERNEL OBJECTS.

INSMOD ALWAYS ACCEPTS ONLY ONE FILENAME AT A TIME.

MODPROBE OFFERS MORE FEATURES THAN INSMOD LIKE IT CAN DECIDE WHICH MODULE IS TO BE LOADED AND IS AWARE OF THE MODULE DEPENDENCIES.

MENTION THE CASE WHEN WE USE “USER VIRTUAL ADDRESS” INSTEAD OF “KERNEL VIRTUAL ADDRESS”?
ANSWER: WHEN WE RUN A PROGRAM IN USERSPACE THEN WE USE “USER VIRTUAL ADDRESS” AS WE DO NOT HAVE ANY ACCESS TO KERNEL VIRTUAL MEMORY ADDRESS. NORMALLY WHEN WE ARE RUNNING OUR PROGRAM IN KERNEL MODE THEN WE USE KERNEL ADDRESS BUT IN CASE WE HAVE TO RUN OUR PROGRAM IN KERNEL MODE AND THAT PROGRAM NEEDS AN INTERACTION WITH A USERSPACE THEN WE WILL USE “USER VIRTUAL ADDRESS” AND BE CAREFUL TO FIRST TRANSLATE IT TO USER VIRTUAL ADDRESS.

MENTION THE WAYS TO DEBUG THE KERNEL CODE.
ANSWER: WE CAN DEBUG A KERNEL CODE SIMPLY WITH THE COMMAND PRINTKS. ELSE WE CAN ALSO USE KDB AND KERNEL PROBES. OTHER METHODS ARE:

UML (USER MODE LINUX) – IT IS THE BEST METHOD FOR DEBUGGING BUT IT DOES NOT SUPPORT DEVICE DRIVERS.
KGDB (KERNEL GNU DEBUGGER)
KDUMP TOOLS WHICH ARE USED TO DUMP KERNEL CORES.
WHAT IS THE DEVICE TREE CONCEPT?
ANSWER: THE QUESTIONS BASED ONDEVICE TREE CONCEPT IS COMMONLY ASKEDIN A LINUX INTERVIEW. DEVICE TREE IS A DATA STRUCTURE WHICH IS USED TO REMOVE THE REPETITIVE CODES IN DIFFERENT BOARDS. THEY ARE LOADED IN THE MEMORY WITH THE HELP OF BOOTLOADER TO A BINARY FILE. HERE THE KERNEL IS USED TO SETTLE THE STRUCTURE OF THE DEVICE TREE ON THE BINARY.

AS MUCH KERNEL IS IMPORTANT IN THE LINUX AS AN OPERATING SYSTEM, THE QUESTIONS BASED ON KERNEL ARE EQUALLY IMPORTANT IN THE LINUX INTERVIEW. DEVICE TREE IS AN IMPORTANT CONCEPT IN KERNEL SO INTERVIEWER MAY ASK THIS QUESTION TO CHECK YOUR KNOWLEDGE ABOUT IT.

HOW CAN WE REDUCE THE SIZE OF THE KERNEL?
ANSWER: THERE ARE CODES WHICH ARE UNNECESSARY AND ARE NOT EXECUTED, WE CAN FIND AND DISABLE THEM TO MAKE THE PROCESSING FASTER IN THE PROJECT. THE KERNEL COMES WITH AN EDITOR KNOWN AS “KERNEL’S CONFIGURATION EDITOR” BY WHICH WE CAN REMOVE AND DISABLE CHUNKS OF CODE THAT ARE NOT REQUIRED.

THERE MAY BE THE CODES FOR WHICH THE HARDWARE IS NOT PRESENT IN THE SYSTEM AND YOU HAVE TO MAKE YOUR SYSTEM UNDERSTAND ABOUT WHAT ARE YOUR SYSTEM’S REQUIREMENTS. BELOW ARE SOME GUIDING PRINCIPLES BY WHICH YOU CAN FIND THE CODES TO BE REMOVED.

HARDWARE NETWORKING DRIVERS:SEVERAL OF SYSTEM-ON-CHIPS HAVE WI-FI DRIVERS, SERIAL AND OTHER HARDWARE THAT ARE NOT USED, YOU CAN REMOVE THOSE DRIVERS THAT ARE BUILT ON THE KERNEL.
FILE SYSTEMS:THE SYSTEM HAS THE ONLY REQUIREMENT OF FEW FILE SYSTEMS BUT IN THE

KERNEL YOU WILL FIND MANY FILE SYSTEMS DRIVERS THAT ARE NOT IN USE E.G. DEVICES WHICH MAKE USE OF FLASH FILE SYSTEMS DO NOT REQUIRE EXT2 OR EXT3 FILE SYSTEM SO THEY CAN BE REMOVED. BE CAUTIOUS THAT DO NOT REMOVE THE FILE SYSTEMS THAT ARE ESSENTIAL OR YOU MAY HAVE THE USE OF THE SYSTEMS IN THE FUTURE.
DEBUGGING AND PROFILING:ALL THE SYSTEMS WHICH COME UNDER KERNEL HACKING ENTRY COULD BE DISABLED IF NOT IN USE.
WHAT ARE KERNEL MODULES IN LINUX?
ANSWER: THE KERNEL MODULES ARE THE SET OF PROGRAMS OR CODE WHICH CAN BE LOADED AS PER THE REQUIREMENT OR DEMAND WHICH CAN BE IMPLEMENTED WITHOUT THE PROCESS OF REBOOTING THE SYSTEM. EACH AND EVERY KERNEL IS A MODULE AND IS EASILY LOADABLE. THERE WILL ALSO BE AN AUTOMATIC MODULE HANDLING.

COMMAND BASED
WHAT IS THE DIFFERENCE BETWEEN “RM” AND “RM –R”?
ANSWER: **“**RM” COMMAND IS USED TO DELETE ALL THE FILES WHILE “RM –R” COMMAND IS USED TO DELETE ALL THE FILES IN A DIRECTORY AND ALSO IN SUBDIRECTORIES.

FOR EXAMPLE,

RM FILE.TXT: IT WILL DELETE THE FILE WITH NAME FILE.TXT

RM –R DIRECTORY: IT WILL REMOVE DIRECTORIES AND SUBDIRECTORIES AND ALSO THEIR CONTENTS.

YOU RUN A BASH SCRIPT AND YOU WANT TO SEE ITS OUTPUT ON YOUR TERMINAL AND SAVE IT TO A FILE AT THE SAME TIME. HOW COULD YOU DO IT?

```
USER@UNKNOWN:~$ SUDO COMMAND -OPTION | TEE LOG.TXT
```

EXPLAIN WHAT ECHO "1" > /PROC/SYS/NET/IPV4/IP_FORWARD DOES.
ENABLE IP FORWARDING ON THE FLY

EXPLAIN THE COMMAND AND METHOD TO CHANGE THE FILE PERMISSIONS IN LINUX.
ANSWER: CHMOD COMMAND IS USED TO CHANGE THE PERMISSIONS OF A FILE. THERE ARE THREE PARTS TO CONSIDER TO SET THE FILE PERMISSIONS.

USER (OR OWNER)
GROUP
OTHER
3 TYPES OF FILE PERMISSION THAT IS GIVEN TO A FILE.

R – READING PERMISSION
W – WRITING PERMISSION
X – EXECUTION PERMISSION
FOR EXAMPLE, CHMOD 751 FILENAME

THEN, THREE NUMBER 751 DESCRIBES PERMISSIONS GIVEN TO THE USER, GROUP AND OTHER IN THE ORDER. EACH NUMBER IS THE SUM OF THE VALUES,I.E. 4 FOR READING, 2 FOR WRITE, 1 FOR EXECUTE.

HERE 751 IS THE COMBINATION OF (4+2+1), (4+0+1), (0+0+1).

SO, CHMOD 751 FILENAME WILL PROVIDE READ, WRITE AND EXECUTE PERMISSION TO THE OWNER; READ AND EXECUTE PERMISSION TO THE GROUP AND ONLY EXECUTE PERMISSION TO THE OTHERS.

HOW CAN WE EDIT A FILE WITHOUT OPENING IN LINUX?
ANSWER: SED COMMAND IS USED TO EDIT A FILE WITHOUT OPENING. SED IS THE ACRONYM FOR STREAMEDITOR. THE “SED” COMMAND IS USED TO MODIFY OR CHANGE THE CONTENTS OF A FILE

FOR EXAMPLE, WE HAVE A TEXT FILE WITH BELOW CONTENT

```
>CAT FILE.TXT
```

WE WANT TO REPLACE THE CONTENT OF THE FILE AND WE WANT TO REPLACE “SED” WITH “VI”. SO, WE WILL USE BELOW COMMAND FOR THIS.

```
>SED 'S/SED/VI/' FILE.TXT
VI COMMAND IS USED TO EDIT A FILE.
```

SO, SED IS REPLACED WITH VI IN THE TEXT.

EXPLAIN GREP COMMAND AND ITS USE.
ANSWER: GREP COMMAND IN LINUX IS USED TO SEARCH A SPECIFIC PATTERN. GREP COMMAND WILL HELP YOU TO EXPLORE THE STRING IN A FILE OR MULTIPLE FILES.

THE SYNTAX FOR GREP COMMAND:

```
GREP ‘WORD’ FILENAME

GREP ‘WORD’ FILE1 FILE2 FILE3

COMMAND|GREP‘STRING’
```

FOR EXAMPLE,

```
GREP “SMITH” PASSWD

GREP “SMITH” PASSWD SHADOW

NETSTAT -AN | GREP8083

CAT /ETC/PASSWD | GREP SMITH
```

EXPLAIN FILE CONTENT COMMANDS ALONG WITH THE DESCRIPTION
ANSWER: THERE ARE MANY COMMANDS PRESENT IN LINUX WHICH ARE USED TO LOOK AT THE CONTENTS OF THE FILE.

HEAD: TO CHECK THE STARTING OF A FILE.

TAIL: TO CHECK THE ENDING OF THE FILE. IT IS THE REVERSE OF HEAD COMMAND.

CAT: USED TO VIEW, CREATE, CONCATENATE THE FILES.

RREP: USED TO FIND THE SPECIFIC PATTERN OR STRING IN A FILE.

MORE: USED TO DISPLAY THE TEXT IN THE TERMINAL WINDOW IN PAGER FORM.

LESS: USED TO VIEW THE TEXT IN THE BACKWARD DIRECTION AND ALSO PROVIDES SINGLE LINE MOVEMENT.

EXPLAIN “CD” COMMAND IN LINUX.
ANSWER: IN LINUX, WHEN A USER NEEDS TO CHANGE THE CURRENT DIRECTORY THEN “CD” COMMAND IS INPUT IN THE SHELL.

SYNTAX: $CD

THE PURPOSE THAT CAN BE FULFILLED BY THE CURRENT COMMAND ARE –

REDIRECT TO A NEW DIRECTORY FROM THE CURRENT DIRECTORY.
CHANGE A DIRECTORY USING ABSOLUTE PATH AND RELATIVE PATH.

THE FOLLOWING COMMANDS ARE UNDER THE CD:

CD ~: REDIRECT TO HOME DIRECTORY.
CD-: REDIRECT TO PREVIOUS DIRECTORY.
CD/: REDIRECT TO ENTIRE SYSTEM DIRECTORY.
MENTION SOME OF THE NETWORKING COMMANDS IN LINUX.
ANSWER: IF YOU CONNECT A SYSTEM TO A NETWORK THEN YOU CAN EASILY TROUBLESHOOT THE CONNECTION ISSUES RELATED TO THE SYSTEM. BELOW ARE FEW OF THE NETWORKING COMMANDS USED FOR CONFIGURATION AND TROUBLESHOOTING.

IFCONFIG(NOW IS IP SOMETHING)
TRACEROUTE
DIG
TELNET
NSLOOKUP
NETSTAT
WRITE THE COMMAND TO VIEW AN EXISTING TAR ARCHIVE AND HOW TO EXTRACT IT?
ANSWER: THE COMMAND FOR VIEWING TAR ARCHIVE THAT IS ALREADY EXISTING: $ TAR TVF ARCHIVE_NAME.TAR

THE COMMAND TO EXTRACT AN EXISTING TAR ARCHIVE: $ TAR XVF ARCHIVE_NAME.TAR

THE COMMAND FOR THE CREATION OF NEW TAR ARCHIVE: $ TAR CVF ARCHIVE_NAME.TAR DIRNAME/

YOU MAY BE ASKED ONE OR MORE COMMAND BASED INTERVIEW QUESTIONS IN THE LINUX INTERVIEW. YOU SHOULD PREPARE YOURSELF WITH AS MANY COMMANDS AS YOU CAN. THERE ARE SEVERAL COMMANDS THAT ARE USED FOR TAR ARCHIVE WHICH ARE COMMONLY ASKED IN THE LINUX INTERVIEW, SO DON'T MISS TO COVER THIS QUESTION WHILE GOING FOR THE LINUX INTERVIEW.

WRITE THE STEPS TO MAKE A USB BOOTABLE DEVICE.
ANSWER: THIS TYPE OF QUESTIONS ARE MOST COMMON IN A LINUX INTERVIEW. FOLLOWINGS ARE THE STEPS TO MAKE A USB BOOTABLE DEVICE –

YOU HAVE TO WRITE EFIDISK.IMG FROM RHEL 6 DVD IMAGES/ SUBDIRECTORY
TO USB DD IF=EFIDISK.IMG OF=/DEV/USB (NAME OF THE USB DEVICE)

NOW YOU HAVE TO DISABLE PING TO AVOID NETWORK /ICMP FLOOD
NOW SET THE FOLLOWING IN/ETC/SYSCTL.CONF : NET.IPV4.ICMP_ECHO_IGNORE_ALL =1
THEN “SYSCTL -P”
MENTION THE STEPS TO CREATE THE PARTITION FORM A RAW DISK.
ANSWER: IN CASE WE WANT TO CREATE A NEW PARTITION FORM A RAW DISK, YOU HAVE TO USE A TOOL KNOWN AS “FDISK UTILITY”. THE STEPS TO CREATE A RAW DISK ARE AS FOLLOWS:

IN CASE OF IDE WE USE >>> FDISK/DEV/HD AND IN CASE OF SCSI WE USE >>> FDISK/DEV/SD
THEN TYPE N FOR CREATING A NEW PARTITION.
AFTER THE PARTITION IS CREATED TYPE ‘W’.
WHAT IS THE COMMAND USED TO GET A GUIDE ON HOW TO USE A COMMAND?
ANSWER: MANUAL PAGES ARE WHERE AN EXPLANATION OF EVERY COMMAND HAS STORED. MANUAL PAGES FOR A SPECIFIC COMMAND WILL HAVE ALL INFORMATION ABOUT THAT COMMAND AND IT CAN BE CALLED AS ‘MAN EG: *‘MAN LS’.*MANUAL PAGES ARE CATEGORIZED INTO DIFFERENT SETS OF USER COMMANDS, SYSTEM CALLS, LIBRARY FUNCTIONS..ETC.. A GENERAL LAYOUT OF A MANUAL PAGE IS –

NAME

THE NAME OF THE COMMAND OR FUNCTION AND SIMPLE EXPLANATION OF IT.

SYNOPSIS

FOR COMMANDS HOW TO RUN IT AND PARAMETERS IT TAKES. FOR FUNCTIONS, A LIST OF THE PARAMETERS IT TAKES AND WHICH HEADER FILE CONTAINS ITS DEFINITION.

DESCRIPTION

A DETAILED DESCRIPTION OF COMMAND OR FUNCTION WE ARE SEARCHING FOR.

EXAMPLES

SOME EXAMPLES OF USAGES.MOST HELPFUL SECTION

SEE ALSO

THIS SECTION WILL HAVE A LIST OF RELATED COMMANDS OR FUNCTIONS.

HOW TO GET A LIST OF CURRENTLY RUNNING PROCESSES AND RESOURCE UTILIZATION IN LINUX?
THE TOP IS THE COMMAND USED FOR THIS. THIS WILL GIVE ALL INFORMATION ABOUT EACH PROCESS RUNNING ON A MACHINE LIKE –

PROCESS ID (PID)
OWNER OF THE PROCESS(USER)
PRIORITY OF PROCESS(PR)
PERCENTAGE OF CPU (%CPU)
PERCENTAGE OF MEMORY
TOTAL CPU TIME SPENDS ON THE PROCESS.
COMMAND USED TO START A PROCESS.
THE POPULAR OPTION USED WITH TOP COMMAND

TOP -U -> PROCESS BY A USER.
TOP – I -> EXCLUDE IDLE TASKS
TOP -P -> SHOW A PARTICULAR PROCESS
WHAT IS A PIPELINE OPERATOR IN LINUX?
ANSWER: PIPELINE OPERATOR IN LINUX IS USED TO REDIRECT THE OUTPUT OF ONE PROGRAM OR COMMAND TO ANOTHER PROGRAM/COMMAND FOR FURTHER PROCESSING. USUALLY TERMED AS REDIRECTION. VERTICAL BARS,’|’ (“PIPES” IN COMMON UNIX VERBIAGE) ARE USED FOR THIS. FOR EXAMPLE, LS -L | GREP KEY, WILL REDIRECT THE OUTPUT OF LS -L COMMAND TO GREP KEY COMMAND

WHAT ARE REGULAR EXPRESSIONS(REGEX)? WHAT IS THE MEANING OF *,+,? IN REGULAR EXPRESSION?
ANSWER: A REGULAR EXPRESSION (REGEX) IS A STRING THAT EXPRESSES A PATTERN USED TO MATCH AGAINST OTHER STRINGS. THE PATTERN WILL EITHER MATCH SOME PORTION OF ANOTHER STRING OR NOT. THERE IS A LIST OF PREDEFINED METACHARACTERS USED IN A REGEX.

* USED TO MATCH THE PRECEDING CHARACTER IF IT APPEARS 0 OR MORE TIMES
+USED TO MATCH THE PRECEDING CHARACTER IF IT APPEARS 1 OR MORE TIMES
? USED TO MATCH THE PRECEDING CHARACTER IF IT APPEARS 0 OR 1 TIMES
WHAT IS A SED COMMAND?
ANSWER: THIS IS THE POPULAR LINUX INTERVIEW QUESTIONS ASKED IN AN INTERVIEW. SED IS A STREAM EDITOR. A STREAM EDITOR IS A PROGRAM THAT TAKES A STREAM OF TEXT AND MODIFIES IT. WITH SED, YOU SPECIFY A REGULAR EXPRESSION WHICH REPRESENTS A PATTERN OF WHAT YOU WANT TO REPLACE. THE GENERIC FORM OF A SED COMMAND IS SED ‘S/PATTERN/REPLACEMENT/’ FILENAME.

WHAT IS UMASK AND WHAT IS ITS USE IN LINUX?
ANSWER: THE UMASK IS A COMMAND WHICH IS OFTEN CALLED AS USER FILE CREATION MASK WHICH IS USED TO CREATE FILE MASK FOR THE USER THAT DETERMINES WHICH FILE OR

DIRECTORY PERMISSIONS ARE AVAILABLE FOR THE USER BASED ON THE READ OR WRITE OR MODIFY TYPE.

NETWORKING:
EXPLAIN, IN AS MUCH DETAIL AS YOU FEEL COMFORTABLE WITH, WHAT IS HAPPENING WHEN YOU ACCESS GOOGLE.COM
YOU TYPE MAPS.GOOGLE.COM INTO THE ADDRESS BAR OF YOUR BROWSER.
THE BROWSER CHECKS THE CACHE FOR A DNS RECORD TO FIND THE CORRESPONDING IP ADDRESS OF MAPS.GOOGLE.COM.
IF THE REQUESTED URL IS NOT IN THE CACHE, ISP’S DNS SERVER INITIATES A DNS QUERY TO FIND THE IP ADDRESS OF THE SERVER THAT HOSTS MAPS.GOOGLE.COM.
BROWSER INITIATES A TCP CONNECTION WITH THE SERVER.
THE BROWSER SENDS AN HTTP REQUEST TO THE WEB SERVER.
THE SERVER HANDLES THE REQUEST AND SENDS BACK A RESPONSE.
THE SERVER SENDS OUT AN HTTP RESPONSE.
THE BROWSER DISPLAYS THE HTML CONTENT (FOR HTML RESPONSES WHICH IS THE MOST COMMON).
WHAT IS SSH PORT FORWARDING?
SSH PORT FORWARDING IS A MECHANISM IN SSH FOR TUNNELING APPLICATION PORTS FROM THE CLIENT MACHINE TO THE SERVER MACHINE, OR VICE VERSA. IT CAN BE USED FOR ADDING ENCRYPTION TO LEGACY APPLICATIONS, GOING THROUGH FIREWALLS, AND SOME SYSTEM ADMINISTRATORS AND IT PROFESSIONALS USE IT FOR OPENING BACKDOORS INTO THE INTERNAL NETWORK FROM THEIR HOME MACHINES. IT CAN ALSO BE ABUSED BY HACKERS AND MALWARE TO OPEN ACCESS FROM THE INTERNET TO THE INTERNAL NETWORK.

HOW TO KNOW WHICH PROCESS LISTENS ON A SPECIFIC PORT?
USING NETSTAT COMMAND:

NETSTAT (NETWORK STATISTICS) COMMAND IS USED TO DISPLAY INFORMATION CONCERNING NETWORK CONNECTIONS, ROUTING TABLES, INTERFACE STATS AND BEYOND.

USE IT WITH GREP COMMAND TO FIND THE PROCESS OR SERVICE LISTENING ON A PARTICULAR PORT IN LINUX AS FOLLOWS (SPECIFY THE PORT).

```
$ NETSTAT -LTNP | GREP -W ':80'
```

CHECK PORT USING NETSTAT COMMAND

CHECK PORT USING NETSTAT COMMAND

IN THE ABOVE COMMAND, THE FLAGS.

L – TELLS NETSTAT TO ONLY SHOW LISTENING SOCKETS.
T – TELLS IT TO DISPLAY TCP CONNECTIONS.
N – INSTRUCTS IT SHOW NUMERICAL ADDRESSES.
P – ENABLES SHOWING OF THE PROCESS ID AND THE PROCESS NAME.
GREP -W – SHOWS MATCHING OF EXACT STRING (:80).
WHAT IS THE DIFFERENCE BETWEEN LOCAL AND REMOTE PORT FORWARDING?
LOCAL FORWARDING IS USED TO FORWARD A PORT FROM THE CLIENT MACHINE TO THE SERVER MACHINE. BASICALLY, THE SSH CLIENT LISTENS FOR CONNECTIONS ON A CONFIGURED PORT, AND WHEN IT RECEIVES A CONNECTION, IT TUNNELS THE CONNECTION TO AN SSH SERVER. THE SERVER CONNECTS TO A CONFIGURATED DESTINATION PORT, POSSIBLY ON A DIFFERENT MACHINE THAN THE SSH SERVER.

TYPICAL USES FOR LOCAL PORT FORWARDING INCLUDE:

TUNNELING SESSIONS AND FILE TRANSFERS THROUGH JUMP SERVERS
CONNECTING TO A SERVICE ON AN INTERNAL NETWORK FROM THE OUTSIDE
CONNECTING TO A REMOTE FILE SHARE OVER THE INTERNET
IN OPENSSH, REMOTE SSH PORT FORWARDINGS ARE SPECIFIED USING THE -R OPTION. FOR

EXAMPLE:

```
SSH -R 8080:LOCALHOST:80 PUBLIC.EXAMPLE.COM
```

THIS ALLOWS ANYONE ON THE REMOTE SERVER TO CONNECT TO TCP PORT 8080 ON THE REMOTE SERVER. THE CONNECTION WILL THEN BE TUNNELED BACK TO THE CLIENT HOST, AND THE CLIENT THEN MAKES A TCP CONNECTION TO PORT 80 ON LOCALHOST. ANY OTHER HOST NAME OR IP ADDRESS COULD BE USED INSTEAD OF LOCALHOST TO SPECIFY THE HOST TO CONNECT TO.

THIS PARTICULAR EXAMPLE WOULD BE USEFUL FOR GIVING SOMEONE ON THE OUTSIDE ACCESS TO AN INTERNAL WEB SERVER.

WHAT IS DNS?
THE DOMAIN NAME SYSTEM RESOLVES THE NAMES OF INTERNET SITES WITH THEIR UNDERLYING IP ADDRESSES ADDING EFFICIENCY AND EVEN SECURITY IN THE PROCESS.

DNS IS A DIRECTORY OF NAMES THAT MATCH WITH NUMBERS. THE NUMBERS, IN THIS CASE ARE IP ADDRESSES, WHICH COMPUTERS USE TO COMMUNICATE WITH EACH OTHER.

WHAT IS AN A RECORD, AN NS RECORD, A PTR RECORD, A CNAME RECORD, AN MX RECORD?
THESE ARE RECORD TYPES THAT ARE PRESENT INSIDE A DNS SERVER

DNS RESOURCE RECORDS (FROM: A, AAAA, CNAME, MX, NS, PTR, SOA, SRV, TXT

ZONE DNS DATABASE IS A COLLECTION OF RESOURCE RECORDS AND EACH OF THE RECORDS PROVIDES INFORMATION ABOUT A SPECIFIC OBJECT. A LIST OF MOST COMMON RECORDS IS PROVIDED BELOW:

ADDRESS MAPPING RECORDS (A)THE RECORD A SPECIFIES IP ADDRESS (IPV4) FOR GIVEN HOST. A RECORDS ARE USED FOR CONVERSION OF DOMAIN NAMES TO CORRESPONDING IP ADDRESSES. IP VERSION 6 ADDRESS RECORDS(AAAA)THE RECORD AAAA (ALSO QUAD-A RECORD) SPECIFIES IPV6 ADDRESS FOR GIVEN HOST. SO IT WORKS THE SAME WAY AS THE A RECORD AND THE DIFFERENCE IS THE TYPE OF IP ADDRESS.
CANONICAL NAME RECORDS(CNAME) THE CNAME RECORD SPECIFIES A DOMAIN NAME THAT HAS TO BE QUERIED IN ORDER TO RESOLVE THE ORIGINAL DNS QUERY. THEREFORE CNAME RECORDS ARE USED FOR CREATING ALIASES OF DOMAIN NAMES. CNAME RECORDS ARE TRULY USEFUL WHEN WE WANT TO ALIAS OUR DOMAIN TO AN EXTERNAL DOMAIN. IN OTHER CASES WE CAN REMOVE CNAME RECORDS AND REPLACE THEM WITH A RECORDS AND EVEN DECREASE PERFORMANCE OVERHEAD
HOST INFORMATION RECORDS(HINFO) ARE USED TO ACQUIRE GENERAL INFORMATION ABOUT A HOST. THE RECORD SPECIFIES TYPE OF CPU AND OS. THE HINFO RECORD DATA PROVIDES THE POSSIBILITY TO USE OPERATING SYSTEM SPECIFIC PROTOCOLS WHEN TWO HOSTS WANT TO COMMUNICATE. FOR SECURITY REASONS THE HINFO RECORDS ARE NOT TYPICALLY USED ON PUBLIC SERVERS.
INTEGRATED SERVICES DIGITAL NETWORK RECORDS (ISDN)THE ISDN RESOURCE RECORD SPECIFIES ISDN ADDRESS FOR A HOST. AN ISDN ADDRESS IS A TELEPHONE NUMBER THAT CONSISTS OF A COUNTRY CODE, A NATIONAL DESTINATION CODE, A ISDN SUBSCRIBER NUMBER AND, OPTIONALLY, A ISDN SUB ADDRESS. THE FUNCTION OF THE RECORD IS ONLY VARIATION OF THE A RESOURCE RECORD FUNCTION.
MAIL EXCHANGER RECORD (MX)THE MX RESOURCE RECORD SPECIFIES A MAIL EXCHANGE SERVER FOR A DNS DOMAIN NAME. THE INFORMATION IS USED BY SIMPLE MAIL TRANSFER PROTOCOL (SMTP) TO ROUTE EMAILS TO PROPER HOSTS. TYPICALLY, THERE ARE MORE THAN ONE MAIL EXCHANGE SERVER FOR A DNS DOMAIN AND EACH OF THEM HAVE SET PRIORITY.
NAME SERVER RECORDS (NS)THE NS RECORD SPECIFIES AN AUTHORITATIVE NAME SERVER FOR GIVEN HOST.
REVERSE-LOOKUP POINTER RECORDS (PTR)AS OPPOSED TO FORWARD DNS RESOLUTION (A AND AAAA DNS RECORDS), THE PTR RECORD IS USED TO LOOK UP DOMAIN NAMES BASED ON AN IP ADDRESS
START OF AUTHORITY RECORDS (SOA) THE RECORD SPECIFIES CORE INFORMATION ABOUT A DNS ZONE, INCLUDING THE PRIMARY NAME SERVER, THE EMAIL OF THE DOMAIN ADMINISTRATOR, THE

DOMAIN SERIAL NUMBER, AND SEVERAL TIMERS RELATING TO REFRESHING THE ZONE.
TEXT RECORDS (TXT): THE TEXT RECORD CAN HOLD ARBITRARY NON-FORMATTED TEXT STRING. TYPICALLY, THE RECORD IS USED BY [ENDER POLICY FRAMEWORK (SPF) TO PREVENT FAKE EMAILS TO APPEAR TO BE SENT BY YOU.
WHAT IS A SPLIT-HORIZON DNS?
SPLIT-BRAIN DNS, SPLIT-HORIZON DNS, OR SPLIT DNS ARE TERMS USED TO DESCRIBE WHEN TWO ZONES FOR THE SAME DOMAIN ARE CREATED, ONE TO BE USED BY THE INTERNAL NETWORK, THE OTHER USED BY THE EXTERNAL NETWORK (USUALLY THE INTERNET). I PREFER THE TERM "SPLIT DNS" SO WE WILL JUST CONTINUE WITH THAT ONE.

A SPLIT DNS INFRASTRUCTURE IS USED TO DIRECT INTERNAL HOSTS TO AN INTERNAL DOMAIN NAME SERVER FOR NAME RESOLUTION AND EXTERNAL HOSTS TO AN EXTERNAL DOMAIN NAME SERVER FOR NAME RESOLUTION. THIS TYPE OF DNS CONFIGURATION IS VERY COMMON IN NETWORKS THAT HAVE ESTABLISHED AN INTERNAL ACTIVE DIRECTORY DOMAIN NAME WHICH IS THE SAME AS THE PUBLIC EXTERNAL DOMAIN NAME. LET'S BEGIN BY TAKING A LOOK AT AN EXAMPLE WHERE SPLIT DNS IS NOT USED.

WHAT IS HTTP?
STANDS FOR "HYPERTEXT TRANSFER PROTOCOL." HTTP IS THE PROTOCOL USED TO TRANSFER DATA OVER THE WEB. IT IS PART OF THE INTERNET PROTOCOL SUITE AND DEFINES COMMANDS AND SERVICES USED FOR TRANSMITTING WEBPAGE DATA.

WHAT IS AN HTTP PROXY AND HOW DOES IT WORK?
AN HTTP PROXY SERVES TWO INTERMEDIARY ROLES AS AN HTTP CLIENT AND AN HTTP SERVER FOR SECURITY, MANAGEMENT, AND CACHING FUNCTIONALITY. THE HTTP PROXY ROUTES HTTP CLIENT REQUESTS FROM A WEB BROWSER TO THE INTERNET, WHILE SUPPORTING THE CACHING OF INTERNET DATA.

PROXY SERVER ADVANTAGES INCLUDE:

MAINTAINING IDENTITY ANONYMITY AS A SECURITY PRECAUTION.
ACCELERATING CACHING RATES.
FACILITATING ACCESS TO PROHIBITED SITES.
ENFORCING ACCESS POLICIES ON CERTAIN WEBSITES.
ALLOWING A SITE TO MAKE EXTERNAL SERVER REQUESTS.
AVOIDING SECURITY CONTROLS.
BYPASSING INTERNET FILTERING FOR ACCESS TO PROHIBITED CONTENT.
DESCRIBE BRIEFLY HOW HTTPS WORKS.
HTTP USES A SERVER-CLIENT MODEL.

WHEN YOU ACCESS A WEBSITE, YOUR BROWSER SENDS A REQUEST TO THE CORRESPONDING WEB SERVER AND IT RESPONDS WITH AN HTTP STATUS CODE. IF THE URL IS VALID AND THE CONNECTION IS GRANTED, THE SERVER WILL SEND YOUR BROWSER THE WEBPAGE AND RELATED FILES.

SOME COMMON HTTP STATUS CODES INCLUDE:

200 - SUCCESSFUL REQUEST (THE WEBPAGE EXISTS)
301 - MOVED PERMANENTLY (OFTEN FORWARDED TO A NEW URL)
401 - UNAUTHORIZED REQUEST (AUTHORIZATION REQUIRED)
403 - FORBIDDEN (ACCESS IS NOT ALLOWED TO THE PAGE OR DIRECTORY)
500 - INTERNAL SERVER ERROR (OFTEN CAUSED BY AN INCORRECT SERVER CONFIGURATION)
HTTP ALSO DEFINES COMMANDS SUCH AS GET AND POST, WHICH ARE USED TO HANDLE FORM SUBMISSIONS ON WEBSITES. THE CONNECT COMMAND IS USED TO FACILITATE A SECURE CONNECTION THAT IS ENCRYPTED USING SSL. ENCRYPTED HTTP CONNECTIONS TAKE PLACE OVER HTTPS, AN EXTENSION OF HTTP DESIGNED FOR SECURE DATA TRANSMISSIONS.

WHAT IS SNMP AND WHAT IS IT USED FOR?
SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL IS A WIDELY USED PROTOCOL FOR MONITORING

THE HEALTH AND WELFARE OF NETWORK EQUIPMENT (EG. ROUTERS), COMPUTER EQUIPMENT AND EVEN DEVICES LIKE UPSS.

IT IS COMMONLY USED BY NETWORK AND SYSTEM ADMINISTRATORS TO GATHER OPERATIONAL STATISTICS(SUCH AS MEASURING NETWORK BANDWIDTH TRAFFIC, CPU USAGE, OR AVAILABLE HARD DRIVE SPACE) AS WELL AS SETTING SYSTEM PARAMETERS.

WHAT IS SMTP? GIVE THE BASIC SCENARIO OF HOW A MAIL MESSAGE IS DELIVERED VIA SMTP.
SMTP STANDS FOR SIMPLE TRANSFER EMAIL PROTOCOL. CURRENTLY, THE ELECTRONIC MAIL (E-MAIL) STANDARD FOR THE INTERNET IS SMTP. SMTP IS THE APPLICATION LEVEL PROTOCOL THAT HANDLES MESSAGE SERVICES OVER [TCP/IP]. SMTP USES TCP WELL KNOWN PORT25.

SIMPLE MAIL TRANSFER PROTOCOL (SMTP) IS BASED ON END-TO-END MESSAGE DELIVERY. AN SIMPLE MAIL TRANSFER PROTOCOL (SMTP) CLIENT CONTACTS THE DESTINATION HOST'S SIMPLE MAIL TRANSFER PROTOCOL (SMTP) SERVER ON WELL-KNOWN PORT 25, TO DELIVER THE MAIL. THE CLIENT THEN WAITS FOR THE SERVER TO SEND A 220 READY FOR MAIL MESSAGE. UPON RECEIPT OF THE 220 MESSAGE, THE CLIENT SENDS A HELO COMMAND. THE SERVER THEN RESPONDS WITH A "250 REQUESTED MAIL ACTION OKAY" MESSAGE.

AFTER THIS, THE MAIL TRANSACTION WILL BEGIN WITH A MAIL COMMAND THAT GIVES THE SENDER IDENTIFICATION AS WELL AS A FROM: FIELD THAT CONTAINS THE ADDRESS TO WHICH ERRORS SHOULD BE REPORTED.

AFTER A SUCCESSFUL MAIL COMMAND, THE SENDER ISSUES A SERIES OF RCPT COMMANDS THAT IDENTIFY RECIPIENTS OF THE MAIL MESSAGE. THE RECEIVER WILL THE ACKNOWLEDGE EACH RCPT COMMAND BY SENDING 250 OK OR BY SENDING THE ERROR MESSAGE 550 NO SUCH USER HERE.

AFTER ALL RCPT COMMANDS HAVE BEEN ACKNOWLEDGED, THE SENDER ISSUES A DATA COMMAND TO INFORM THE RECEIVER THAT THE SENDER IS READY TO TRANSFER A COMPLETE MAIL MESSAGE. THE RECEIVER RESPONDS WITH MESSAGE 354 START MAIL COMMAND WITH AN ENDING SEQUENCE THAT THE SENDER SHOULD USE TO TERMINATE THE MESSAGE DATA. THE TERMINATION SEQUENCE CONSISTS OF 5 CHARACTERS: CARRIAGE RETURN, LINE FEED, PERIOD, CARRIAGE RETURN, AND LINE FEED (.).

THE CLIENT NOW SENDS THE DATA LINE BY LINE, ENDING WITH THE 5-CHARACTER SEQUENCE . LINE, UPON WHICH THE RECEIVER WILL ACKNOWLEDGE WITH A 250 OK, OR AN APPROPRIATE ERROR MESSAGE IF ANYTHING WENT WRONG.

AFTER THE SENDING IS COMPLETED, THE CLIENT CAN FOLLOW ANY OF THESE ACTIONS.

WHAT IS LOCALHOST AND WHY WOULD PING LOCALHOST FAIL?
IN COMPUTER NETWORKING LOCALHOST IS A HOSTNAME THAT MEANS THIS COMPUTER. IT IS USED TO ACCESS THE NETWORK SERVICES THAT ARE RUNNING ON THE HOST VIA THE LOOPBACK NETWORK INTERFACE. USING THE LOOPBACK INTERFACE BYPASSES ANY LOCAL NETWORK INTERFACE.

IF PING LOCALHOST FAIL WE SHOULD SE IF THERE IS AN INTERFACE CONFIGURED WITH LO0 OR ANY OTHER INTERFACE WITH 127.0.0.1? CHECK THE RX PACKETS/TX PACKETS COUNT. ALSO, CHECK TO SEE IF LO0 IS CONFIGURED IN /ETC/NETWORK/INTERFACES.

OUTPUT OF 'IFCONFIG'

```
LO        LINK ENCAP:LOCAL LOOPBACK
          INET ADDR:127.0.0.1  MASK:255.0.0.0
          INET6 ADDR: ::1/128 SCOPE:HOST
          UP LOOPBACK RUNNING  MTU:16436  METRIC:1
          RX PACKETS:24 ERRORS:0 DROPPED:0 OVERRUNS:0 FRAME:0
          TX PACKETS:24 ERRORS:0 DROPPED:0 OVERRUNS:0 CARRIER:0
          COLLISIONS:0 TXQUEUELEN:0
          RX BYTES:1440 (1.4 KB)  TX BYTES:1440 (1.4 KB)
```

WHAT IS THE SIMILARITY BETWEEN "PING" & "TRACEROUTE" ? HOW IS TRACEROUTE ABLE TO FIND THE HOPS.
THE MAIN DIFFERENCE BETWEEN THE COMMON PING AND TRACEROUTE COMMANDS IS THAT PING IS A QUICK AND EASY WAY TO TELL YOU IF THE DESTINATION SERVER IS ONLINE AND ESTIMATES HOW LONG IT TAKES TO SEND AND RECEIVE DATA TO THE DESTINATION. TRACEROUTE TELLS YOU THE EXACT ROUTE YOU TAKE TO REACH THE SERVER FROM YOUR COMPUTER (ISP) AND HOW LONG EACH HOP TAKES.

TRACEROUTE MAKES USE OF A NETWORK MECHANISM CALLED TTL, OR "TIME TO LIVE" AND PROBING THE HOPS: TRACEROUTE MAKES SURE THAT EACH HOP ON THE WAY TO A DESTINATION DEVICE DROPS A PACKET, AND SENDS BACK AN ICMP ERROR MESSAGE.

WHAT IS THE COMMAND USED TO SHOW ALL OPEN PORTS AND/OR SOCKET CONNECTIONS ON A MACHINE?

```
SS -S # LIST CURRENTLY ESTABLISHED, CLOSED, ORPHANED AND WAITING TCP SOCKETS
SS -L # DISPLAY ALL OPEN NETWORK PORTS
SS -PL # TO SEE PROCESS NAMED USING OPEN SOCKET:
 SS -LP | GREP 4949 # FIND OUT WHO IS RESPONSIBLE FOR OPENING SOCKET / PORT # 4949 USING THE SS COMMAND AND GREP COMMAND
```

IS 300.168.0.123 A VALID IPV4 ADDRESS?
A VALID IP ADDRESS MUST BE IN THE FORM OF XXX.XXX.XXX.XXX, WHERE XXX IS A NUMBER FROM 0-255 SO NO, THIS ISN'T A VALID IPV4 ADDRESS.

WHICH IP RANGES/SUBNETS ARE "PRIVATE" OR "NON-ROUTABLE" (RFC 1918)?
A PRIVATE NETWORK IS TYPICALLY A NETWORK THAT USES PRIVATE IP ADDRESS SPACE, FOLLOWING THE RFC 1918]STANDARD

THE CURRENT IANA PRIVATE INTERNET (ALSO CALLED NON-ROUTABLE) ADDRESSES ARE:

NAME	IP ADDRESS RANGE	NUMBER OF ADDRESSES	CLASSFUL DESCRIPTION	LARGEST CIDR BLOCK
24-BIT BLOCK	10.0.0.0 – 10.255.255.255	16,777,216	SINGLE CLASS A, 256 CONTIGUOUS CLASS BS	10.0.0.0/8
20-BIT BLOCK	172.16.0.0 – 172.31.255.255	1,048,576	16 CONTIGUOUS CLASS BS	172.16.0.0/12
16-BIT BLOCK	192.168.0.0 – 192.168.255.255	65,536	SINGLE CLASS B, 256 CONTIGUOUS CLASS CS	192.168.0.0/16

WHAT IS A VLAN?
A VIRTUAL LAN (VLAN) IS ANY BROADCAST DOMAIN THAT IS PARTITIONED AND ISOLATED IN A COMPUTER NETWORK AT THE DATA LINK LAYER. LAN IS THE ABBREVIATION FOR LOCAL AREA NETWORK AND IN THIS CONTEXT VIRTUAL REFERS TO A PHYSICAL OBJECT RECREATED AND ALTERED BY ADDITIONAL LOGIC. VLANS WORK BY APPLYING TAGS TO NETWORK FRAMES AND HANDLING THESE TAGS IN NETWORKING SYSTEMS – CREATING THE APPEARANCE AND FUNCTIONALITY OF NETWORK TRAFFIC THAT IS PHYSICALLY ON A SINGLE NETWORK BUT ACTS AS IF IT IS SPLIT BETWEEN SEPARATE NETWORKS. IN THIS WAY, VLANS CAN KEEP NETWORK APPLICATIONS SEPARATE DESPITE BEING CONNECTED TO THE SAME PHYSICAL NETWORK, AND WITHOUT REQUIRING MULTIPLE SETS OF CABLING AND NETWORKING DEVICES TO BE DEPLOYED.

WHAT IS ARP AND WHAT IS IT USED FOR?
THE ADDRESS RESOLUTION PROTOCOL (ARP) IS A COMMUNICATION PROTOCOL USED FOR DISCOVERING THE LINK LAYER ADDRESS, SUCH AS A MAC ADDRESS, ASSOCIATED WITH A GIVEN INTERNET LAYER ADDRESS, TYPICALLY AN IPV4 ADDRESS.THIS MAPPING IS A CRITICAL FUNCTION IN THE INTERNET PROTOCOL SUITE

WHAT IS THE DIFFERENCE BETWEEN TCP AND UDP?
BOTH TCP AND UDP ARE PROTOCOLS USED FOR SENDING BITS OF DATA — KNOWN AS PACKETS — OVER THE INTERNET. THEY BOTH BUILD ON TOP OF THE INTERNET PROTOCOL. IN OTHER WORDS, WHETHER YOU ARE SENDING A PACKET VIA TCP OR UDP, THAT PACKET IS SENT TO AN IP

ADDRESS. THESE PACKETS ARE TREATED SIMILARLY, AS THEY ARE FORWARDED FROM YOUR COMPUTER TO INTERMEDIARY ROUTERS AND ON TO THE DESTINATION.

TCP (TRANSMISSION CONTROL PROTOCOL) IS CONNECTION ORIENTED, WHEREAS UDP (USER DATAGRAM PROTOCOL) IS CONNECTION-LESS. THIS MEANS THAT TCP TRACKS ALL DATA SENT, REQUIRING ACKNOWLEDGMENT FOR EACH OCTET (GENERALLY).

WHAT IS THE PURPOSE OF A DEFAULT GATEWAY?
A DEFAULT GATEWAY SERVES AS AN ACCESS POINT OR IP ROUTER THAT A NETWORKED COMPUTER USES TO SEND INFORMATION TO A COMPUTER IN ANOTHER NETWORK OR THE INTERNET. DEFAULT SIMPLY MEANS THAT THIS GATEWAY IS USED BY DEFAULT, UNLESS AN APPLICATION SPECIFIES ANOTHER GATEWAY.

A DEFAULT GATEWAY LETS DEVICES IN ONE NETWORK COMMUNICATE WITH DEVICES IN ANOTHER NETWORK. IF YOUR COMPUTER, FOR EXAMPLE, IS REQUESTING AN INTERNET WEB PAGE, THE REQUEST FIRST RUNS THROUGH YOUR DEFAULT GATEWAY BEFORE EXITING THE LOCAL NETWORK TO REACH THE INTERNET.

WHAT ARE THE LAYERS OF THE OSI MODEL?
THE SEVEN LAYERS OF FUNCTION ARE PROVIDED BY A COMBINATION OF APPLICATIONS, OPERATING SYSTEMS, NETWORK CARD DEVICE DRIVERS AND NETWORKING HARDWARE THAT ENABLE A SYSTEM TO TRANSMIT A SIGNAL OVER A NETWORK ETHERNET OR FIBBER OPTIC CABLE OR THROUGH WI-FI OR OTHER WIRELESS PROTOCOLS

THE SEVEN OPEN SYSTEMS INTERCONNECTION LAYERS ARE:

LAYER 7: THE APPLICATION LAYER. THIS IS THE LAYER AT WHICH COMMUNICATION PARTNERS ARE IDENTIFIED -- IS THERE SOMEONE TO TALK TO? -- NETWORK CAPACITY IS ASSESSED -- WILL THE NETWORK LET ME TALK TO THEM RIGHT NOW? -- AND WHERE THE DATA OR APPLICATION IS PRESENTED IN A VISUAL FORM THE USER CAN UNDERSTAND. THIS LAYER IS NOT THE APPLICATION ITSELF, IT IS THE SET OF SERVICES AN APPLICATION SHOULD BE ABLE TO MAKE USE OF DIRECTLY, ALTHOUGH SOME APPLICATIONS MAY PERFORM APPLICATION-LAYER FUNCTIONS.

LAYER 6: THE PRESENTATION LAYER.THIS LAYER IS USUALLY PART OF AN OPERATING SYSTEM OS AND CONVERTS INCOMING AND OUTGOING DATA FROM ONE PRESENTATION [FORMAT TO ANOTHER -- FOR EXAMPLE, FROM CLEAR TEXT TO ENCRYPTED TEXT AT ONE END AND BACK TO CLEAR TEXT AT THE OTHER.

LAYER 5: THE SESSION LAYER. THIS LAYER SETS UP, COORDINATES AND TERMINATES CONVERSATIONS. ITS SERVICES INCLUDE AUTHENTICATION AND RECONNECTION AFTER AN INTERRUPTION. ON THE INTERNET, TRANSMISSION CONTROL PROTOCOL TCP AND USER DATAGRAM PROTOCOL UDP PROVIDE THESE SERVICES FOR MOST APPLICATIONS.

LAYER 4: THE TRANSPORT LAYER THIS LAYER MANAGES PACKETIZATION OF DATA, THEN THE DELIVERY OF THE PACKETS, INCLUDING CHECKING FOR ERRORS IN THE DATA ONCE IT ARRIVES. ON THE INTERNET, TCP AND UDP PROVIDE THESE SERVICES FOR MOST APPLICATIONS AS WELL.

LAYER 3: THE NETWORK LAYER. THIS LAYER HANDLES ADDRESSING AND ROUTING THE DATA -- SENDING IT IN THE RIGHT DIRECTION TO THE RIGHT DESTINATION ON OUTGOING TRANSMISSIONS AND RECEIVING INCOMING TRANSMISSIONS AT THE PACKET LEVEL. IP IS THE NETWORK LAYER FOR THE INTERNET.

LAYER 2: THE DATA-LINK LAYER. THIS LAYER SETS UP LINKS ACROSS THE PHYSICAL NETWORK, PUTTING PACKETS INTO NETWORK FRAMES. THIS LAYER HAS TWO SUB-LAYERS: THE LOGICAL LINK CONTROL LAYER AND THE MEDIA ACCESS CONTROL LAYER MAC. MAC LAYER TYPES INCLUDE ETHERNET AND 802.11 WIRELESS SPECIFICATIONS.

LAYER 1: THE PHYSICAL LAYER. THIS LAYER CONVEYS THE BIT STREAM ACROSS THE NETWORK

EITHER ELECTRICALLY, MECHANICALLY OR THROUGH RADIO WAVES. THE PHYSICAL LAYER COVERS A VARIETY OF DEVICES AND MEDIUMS, AMONG THEM CABLING, CONNECTORS, RECEIVERS, TRANSCEIVERS AND REPEATERS.

DESCRIBE BRIEFLY THE STEPS YOU NEED TO TAKE IN ORDER TO CREATE AND INSTALL A VALID CERTIFICATE FOR THE SITE HTTPS://FOO.EXAMPLE.COM.
HTTPS://WWW.FREECODECAMP.ORG/NEWS/HOW-TO-GET-HTTPS-WORKING-ON-YOUR-LOCAL-DEVELOPMENT-ENVIRONMENT-IN-5-MINUTES-7AF615770EEC/

E FALA DO LET'S ENCRYPT HTTPS://LETSENCRYPT.ORG/

CAN YOU HAVE SEVERAL HTTPS VIRTUAL HOSTS SHARING THE SAME IP?
AS MANY AS YOU WANT. JUST DEFINE VIRTUAL HOSTS AND ASSIGN SSL CERTIFICATES TO THEM AS NEEDED.

WHAT IS A WILDCARD CERTIFICATE?
A SSL WILDCARD CERTIFICATE IS A SINGLE CERTIFICATE WITH A WILDCARD CHARACTER IN THE DOMAIN NAME FIELD. THIS ALLOWS THE CERTIFICATE TO SECURE MULTIPLE SUB DOMAIN NAMES (HOSTS) PERTAINING TO THE SAME BASE DOMAIN.

FOR EXAMPLE, A WILDCARD CERTIFICATE FOR *.(DOMAINNAME).COM, COULD BE USED FOR WWW.(DOMAINNAME).COM, MAIL.(DOMAINNAME).COM, STORE.(DOMAINNAME).COM, IN ADDITION TO ANY ADDITIONAL SUB DOMAIN NAME IN THE (DOMAINNAME).COM.

WHAT IS COMMAND USED TO SHOW THE ROUTING TABLE ON A LINUX BOX?
IP ROUTE #OR IP R
A TCP CONNECTION ON A NETWORK CAN BE UNIQUELY DEFINED BY 4 THINGS. WHAT ARE THOSE THINGS?
THE TCP LAYER ON EITHER END MAINTAINS TABLE ENTRIES CORRESPONDING TO THE 4-TUPLE (REMOTE-IP-ADDRESS, REMOTE-PORT, SOURCE-IP-ADDRESS, SOURCE-PORT). THIS 4-TUPLE UNIQUELY IDENTIFIES A CONNECTION.

WHEN A CLIENT RUNNING A WEB BROWSER CONNECTS TO A WEB SERVER, WHAT IS THE SOURCE PORT AND WHAT IS THE DESTINATION PORT OF THE CONNECTION?
SOURCE PORTS ARE RANDOMLY GENERATED FROM THE UNREGISTERED PORT RANGE.

THE SOURCE/DESTINATION PORT WORKS SIMILAR TO YOUR IP. THE PORT YOU SEND FROM, IS THE PORT THE SERVICE WILL REPLY TOO. FOR INSTANCE; A WEBSITE IS SIMPLY A SERVER LISTENING FOR CONNECTIONS ON PORT 80 (OR 443).

WHEN YOU ATTEMPT TO LOAD A WEBSITE YOU GENERATE A FREE PORT FROM THE UNREGISTERED RANGE AND SEND THE REQUEST FROM 192.168.1.1:45676 (SOURCE PORT SELECTED) YOUR BROWSER THEN SENDS THE REQUEST TO 200.20.20.20:80 (IP IS AN EXAMPLE) AS THE SERVER IS LISTENING ON THIS PORT.

MUCH LIKE YOUR IP, WHEN THE SERVER REPLIES TO YOU IT SETS THE DESTINATION IP AND PORT IN THE PACKETS HEADER TO THE SOURCE IP / PORT IT RECEIVED THE REQUEST ON. THIS ENABLES YOU TO RUN MULTIPLE WEBPAGES AT ONCE.

IF YOU HAD 4 WEBPAGES OPEN, ALL SENDING AND RECEIVING ON PORT 80, IT WOULD NOT KNOW WHERE THE RESPONSE TRAFFIC IS MEANT TO BE DIRECTED. THIS IS WHY THE SOURCE PORT IS USED.

HOW DO YOU ADD AN IPV6 ADDRESS TO A SPECIFIC INTERFACE?
ADDING AN IPV6 ADDRESS IS SIMILAR TO THE MECHANISM OF "IP ALIAS" ADDRESSES IN LINUX IPV4 ADDRESSED INTERFACES.

USAGE:

```
# /SBIN/IP -6 ADDR ADD <IPV6ADDRESS>/<PREFIXLENGTH> DEV <INTERFACE>
```

EXAMPLE:

```
# /SBIN/IP -6 ADDR ADD 2001:0DB8:0:F101::1/64 DEV ETH0
```

YOU HAVE ADDED AN IPV4 AND IPV6 ADDRESS TO INTERFACE ETH0. A PING TO THE V4 ADDRESS IS WORKING BUT A PING TO THE V6 ADDRESS GIVES YOU THE RESPONSE SENDMSG: OPERATION NOT PERMITTED. WHAT COULD BE WRONG?

THIS MEANS THAT YOUR SERVER IS NOT ALLOWED TO SEND ICMP PACKETS.

CHECK FIREWALL RULES:

```
$ IP6TABLES -P INPUT ACCEPT
$ IP6TABLES -P OUTPUT ACCEPT
$ IP6TABLES -P FORWARD ACCEPT
```

HOW MANY NTP SERVERS WOULD YOU CONFIGURE IN YOUR LOCAL NTP.CONF?

IT IS NOT RECOMMENDED TO USE ONLY TWO NTP SERVERS.

IF MORE THAN ONE NTP SERVER IS REQUIRED, FOUR NTP SERVERS IS THE RECOMMENDED MINIMUM. FOUR SERVERS PROTECTS AGAINST ONE INCORRECT TIMESOURCE, OR "FALSETICKER".

WHAT DOES THE COLUMN 'REACH' MEAN IN NTPQ -P OUTPUT?

THE WHEN COLUMN SHOWS THE TIME SINCE THE PEER WAS LAST HEARD IN SECONDS, WHILE THE REACH COLUMN SHOWS THE STATUS OF THE REACHABILITY REGISTER (SEE RFC-1305) IN OCTAL.

YOU NEED TO UPGRADE KERNEL AT 100-1000 SERVERS, HOW YOU WOULD DO THIS?

PUPPET OU HTTPS://WWW.QUORA.COM/YOU-NEED-TO-UPGRADE-KERNEL-AT-100-1000-LINUX-SERVERS-HOW-YOU-WOULD-DO-THIS

HOW CAN YOU TELL IF THE HTTPD PACKAGE WAS ALREADY INSTALLED?

TRY INSTALL IT AGAIN? OR CHECK IT VERSION HTTPD -V

HOW CAN YOU LIST THE CONTENTS OF A PACKAGE?

DPKG -C (OR --CONTENTS) LISTS THE CONTENTS OF A .DEB PACKAGE FILE

HOW CAN YOU DETERMINE WHICH PACKAGE IS BETTER: OPENSSH-SERVER-5.3P1-118.1.EL6_8.X86_64 OR OPENSSH-SERVER-6.6P1-1.EL6.X86_64 ?

ALWAYS MORE RECENT VERSION, SO SECOND ONE.

WHAT IS SNAT AND WHEN SHOULD IT BE USED?

SOURCE NAT: SOURCE NETWORK ADDRESS TRANSLATION DESTINATION NAT: DESTINATION NETWORK ADDRESS TRANSLATION

USE-CASE FOR SOURCE NAT: A LOCAL CLIENT BEHIND FIREWALL OR NAT DEVICE WANTED TO BROWSE INTERNET.

SOURCE NAT (SNAT) IS THE MOST COMMON FORM OF NAT. SNAT CHANGES THE SOURCE ADDRESS OF THE PACKETS PASSING THROUGH THE ROUTER. SNAT IS TYPICALLY USED WHEN AN INTERNAL (PRIVATE) HOST NEEDS TO INITIATE A SESSION TO AN EXTERNAL (PUBLIC) HOST; IN THIS CASE, THE DEVICE THAT IS PERFORMING NAT CHANGES THE PRIVATE IP ADDRESS OF THE SOURCE HOST TO SOME PUBLIC IP ADDRESS, AS SHOWN IN THE FOLLOWING FIGURE. IN “MASQUERADE” NAT (A COMMON TYPE OF SNAT), THE SOURCE ADDRESS OF THE OUTGOING PACKET IS REPLACED WITH THE PRIMARY IP ADDRESS OF THE OUTBOUND INTERFACE. THE DESTINATION ADDRESS OF RETURN PACKETS IS AUTOMATICALLY TRANSLATED BACK TO THE IP ADDRESS OF THE SOURCE HOST.

EXPLAIN HOW COULD YOU SSH LOGIN INTO A LINUX SYSTEM THAT DROPS ALL NEW INCOMING PACKETS USING A SSH TUNNEL.

HTTPS://UNIX.STACKEXCHANGE.COM/QUESTIONS/46235/HOW-DOES-REVERSE-SSH-TUNNELING-WORK

HOW DO YOU STOP A DDOS ATTACK?

NETFILTER IPTABLES (SOON TO BE REPLACED BY NFTABLES) IS A USER-SPACE COMMAND LINE

UTILITY TO CONFIGURE KERNEL PACKET FILTERING RULES DEVELOPED BY NETFILTER.

IT’S THE DEFAULT FIREWALL MANAGEMENT UTILITY ON LINUX SYSTEMS – EVERYONE WORKING WITH LINUX SYSTEMS SHOULD BE FAMILIAR WITH IT OR HAVE AT LEAST HEARD OF IT.

IPTABLES CAN BE USED TO FILTER CERTAIN PACKETS, BLOCK SOURCE OR DESTINATION PORTS AND IP ADDRESSES, FORWARD PACKETS VIA NAT AND A LOT OF OTHER THINGS.

YOU CAN USE IT TO BLOCK THE IP OR IP-RANGES THAT ARE TARGETTING YOUR NETWORK.

YOU CAN ALSO USE COMMERCIAL TOOLS THAT DO IT FOR YOU LIKE CLOUDFLARE.

HOW CAN YOU SEE CONTENT OF AN IP PACKET?

```
# TCPDUMP -R /TMP/CAPTURE -A | GREP '10.2.1.50'
```

-A OPTION TO TCPDUMP GIVES THE PACKET CONTENTS

WHAT IS IPOAC (RFC 1149)?
IP OVER AVIAN CARRIERS (IPOAC) IS A PROPOSAL TO CARRY INTERNET PROTOCOL (IP) TRAFFIC BY BIRDS SUCH AS HOMING PIGEONS.

WHAT WILL HAPPEN WHEN YOU BIND PORT 0?
ASKING TO BIND TCP ON PORT 0 INDICATES A REQUEST TO DYNAMICALLY GENERATE AN UNUSED PORT NUMBER. IN OTHER WORDS, THE PORT NUMBER YOU'RE ACTUALLY LISTENING ON AFTER THAT REQUEST IS NOT ZERO.

WHAT IS THE DIFFERENCE BETWEEN A HUB AND A SWITCH?
A SWITCH IS USED TO CONNECT VARIOUS NETWORK SEGMENTS. A NETWORK SWITCH IS A SMALL HARDWARE DEVICE THAT JOINS MULTIPLE COMPUTERS TOGETHER WITHIN ONE LOCAL AREA NETWORK (LAN). A HUB CONNECTS MULTIPLE ETHERNET DEVICES TOGETHER, MAKING THEM ACT AS A SINGLE SEGMENT.

NMAP COMMAND
NMAP IS USED TO DISCOVER HOSTS AND SERVICES ON A COMPUTER NETWORK BY SENDING PACKETS AND ANALYZING THE RESPONSES. NMAP PROVIDES A NUMBER OF FEATURES FOR PROBING COMPUTER NETWORKS, INCLUDING HOST DISCOVERY AND SERVICE AND OPERATING SYSTEM DETECTION

SHELL SCRIPTING:
WHAT IS THE DIFFERENCE BETWEEN THESE TWO COMMANDS?

```
MYVAR=HELLO

EXPORT MYVAR=HELLO
```

. ###### WHAT IS BASH QUICK SUBSTITUTION/CARET REPLACE(^X^Y)?

HTTPS://STACKOVERFLOW.COM/QUESTIONS/2149482/CARET-SEARCH-AND-REPLACE-IN-BASH-SHELL

IT MIGHT BE EASIER FOR YOU TO REMEMBER THE "LINE NOISE" VERSION IF YOU ALSO THINK OF ^STRING1^STRING2 AS ALREADY BEING EQUIVALENT TO !!:S/STRING1/STRING2/.

WHAT IS A TARPIPE (OR, HOW WOULD YOU GO ABOUT COPYING EVERYTHING, INCLUDING HARDLINKS AND SPECIAL FILES, FROM ONE SERVER TO ANOTHER)?
TARPIPE (OR, HOW WOULD YOU GO ABOUT COPYING EVERYTHING, INCLUDING HARDLINKS AND SPECIAL FILES, FROM ONE SERVER TO ANOTHER)

. ###### WHAT IS A SHELL?

SHELL IS AN INTERFACE BETWEEN THE USER AND THE KERNEL. EVEN THOUGH THERE CAN BE

ONLY ONE KERNEL; A SYSTEM CAN HAVE MANY SHELL RUNNING SIMULTANEOUSLY. SO, WHENEVER A USER ENTERS A COMMAND THROUGH THE KEYBOARD, THE SHELL COMMUNICATES WITH THE KERNEL TO EXECUTE IT AND THEN DISPLAY THE OUTPUT TO THE USER.

. ###### WHAT ARE THE DIFFERENT TYPES OF COMMONLY USED SHELLS ON A TYPICAL LINUX SYSTEM?

CSH,KSH,BASH,BOURNE . THE MOST COMMONLY USED AND ADVANCED SHELL USED TODAY IS "BASH" .

. ###### WHAT IS THE EQUIVALENT OF A FILE SHORTCUT THAT WE HAVE A WINDOW ON A LINUX SYSTEM?**

SHORTCUTS ARE CREATED USING "LINKS" ON LINUX. THERE ARE TWO TYPES OF LINKS THAT CAN BE USED NAMELY "SOFT LINK" AND "HARD LINK".

. ###### WHAT IS THE DIFFERENCE BETWEEN SOFT AND HARD LINKS?

SOFT LINKS ARE LINK TO THE FILE NAME AND CAN RESIDE ON DIFFERENT FILESYTEM AS WELL; HOWEVER HARD LINKS ARE LINK TO THE INODE OF THE FILE AND HAVE TO BE ON THE SAME FILESYTEM AS THAT OF THE FILE. DELETING THE ORIGINAL FILE MAKES THE SOFT LINK INACTIVE (BROKEN LINK) BUT DOES NOT AFFECT THE HARD LINK (HARD LINK WILL STILL ACCESS A COPY OF THE FILE)

HOW WILL YOU PASS AND ACCESS ARGUMENTS TO A SCRIPT IN LINUX?
ARGUMENTS CAN BE PASSED AS: SCRIPTNAME "ARG1" "ARG2"…."ARGN" AND CAN BE ACCESSED INSIDE THE SCRIPT AS $1 , $2 .. $N

. ###### WHAT IS THE SIGNIFICANCE OF $#?

$# SHOWS THE COUNT OF THE ARGUMENTS PASSED TO THE SCRIPT.

WHAT IS THE DIFFERENCE BETWEEN
@?
* WILL CONSIDER THE ENTIRE SET OF POSITIONAL PARAMETERS AS A SINGLE STRING.

8.USE SED COMMAND TO REPLACE THE CONTENT OF THE FILE (EMULATE TAC COMMAND)
EG:

```
IF CAT FILLE
ABCD
EFGH
```
THEN O/P SHOULD BE

```
EFGH ABCD
```

```
SED '1! G; H;$!D' FILE1
```
HERE G COMMAND APPENDS TO THE PATTERN SPACE,

H COMMAND COPIES PATTERN BUFFER TO HOLD BUFFER

AND D COMMAND DELETES THE CURRENT PATTERN SPACE.

9. GIVEN A FILE, REPLACE ALL OCCURRENCE OF WORD "ABC" WITH "DEF" FROM 5TH LINE TILL END IN ONLY THOSE LINES THAT CONTAINS WORD "MNO"
```
SED –N '5,$P' FILE1|SED '/MNO/S/ABC/DEF/'
```
10. GIVEN A FILE, WRITE A COMMAND SEQUENCE TO FIND THE COUNT OF EACH WORD.
```
TR –S  "(BACKSLASH)040" <FILE1|TR –S  "(BACKSLASH)011"|TR "(BACKSLASH)040 (BACKSLASH)011" "(BACKSLASH)012" |UNIQ –C
```

WHERE "(BACKSLASH)040" IS OCTAL EQUIVALENT OF "SPACE"
"(BACKSLASH)011" IS AN OCTAL EQUIVALENT OF "TAB CHARACTER" AND

"(BACKSLASH)012" IS AN OCTAL EQUIVALENT OF THE NEWLINE CHARACTER.

HOW WILL YOU FIND THE 99TH LINE OF A FILE USING ONLY TAIL AND HEAD COMMAND?

```
TAIL +99 FILE1|HEAD -1
```

PRINT THE 10TH LINE WITHOUT USING TAIL AND HEAD COMMAND.

```
SED –N '10P' FILE1
```

IN MY BASH SHELL I WANT MY PROMPT TO BE OF FORMAT '$"PRESENT WORKING DIRECTORY":"HOSTNAME"> AND LOAD A FILE CONTAINING A LIST OF USER-DEFINED FUNCTIONS AS SOON AS I LOG IN, HOW WILL YOU AUTOMATE THIS?
IN BASH SHELL, WE CAN CREATE ".PROFILE" FILE WHICH AUTOMATICALLY GETS INVOKED AS SOON AS I LOG IN AND WRITE THE FOLLOWING SYNTAX INTO IT.

```
EXPORT PS1='$ `PWD`:`HOSTNAME`>' .FILE1
```

HERE FILE1 IS THE FILE CONTAINING THE USER-DEFINED FUNCTIONS AND "." INVOKES THIS FILE IN CURRENT SHELL.

EXPLAIN ABOUT "S" PERMISSION BIT IN A FILE.
"S" BIT IS CALLED "SET USER ID" (SUID) BIT.

"S" BIT ON A FILE CAUSES THE PROCESS TO HAVE THE PRIVILEGES OF THE OWNER OF THE FILE DURING THE INSTANCE OF THE PROGRAM.

FOR EXAMPLE, EXECUTING "PASSWD" COMMAND TO CHANGE CURRENT PASSWORD CAUSES THE USER TO WRITES ITS NEW PASSWORD TO SHADOW FILE EVEN THOUGH IT HAS "ROOT" AS ITS OWNER.

I WANT TO CREATE A DIRECTORY SUCH THAT ANYONE IN THE GROUP CAN CREATE A FILE AND ACCESS ANY PERSON'S FILE IN IT BUT NONE SHOULD BE ABLE TO DELETE A FILE OTHER THAN THE ONE CREATED BY HIMSELF.
WE CAN CREATE THE DIRECTORY GIVING READ AND EXECUTE ACCESS TO EVERYONE IN THE GROUP AND SETTING ITS STICKY BIT "T" ON AS FOLLOWS:

```
MKDIR DIREC1

CHMOD G+WX DIREC1

CHMOD +T DIREC1
```

HOW CAN YOU FIND OUT HOW LONG THE SYSTEM HAS BEEN RUNNING?
WE CAN FIND THIS BY USING THE COMMAND "UPTIME".

HOW CAN ANY USER FIND OUT ALL INFORMATION ABOUT A SPECIFIC USER LIKE HIS DEFAULT SHELL, REAL-LIFE NAME, DEFAULT DIRECTORY, WHEN AND HOW LONG HE HAS BEEN USING THE SYSTEM?
FINGER "LOGINNAME" …WHERE LOGINNAME IS THE LOGIN NAME OF THE USER WHOSE INFORMATION IS EXPECTED.

WHAT IS THE DIFFERENCE BETWEEN $ AND $!?
$$ GIVES THE PROCESS ID OF THE CURRENTLY EXECUTING PROCESS WHEREAS $! SHOWS THE PROCESS ID OF THE PROCESS THAT RECENTLY WENT INTO THE BACKGROUND.

WHAT ARE ZOMBIE PROCESSES?
THESE ARE THE PROCESSES WHICH HAVE DIED BUT WHOSE EXIT STATUS IS STILL NOT PICKED BY THE PARENT PROCESS. THESE PROCESSES EVEN IF NOT FUNCTIONAL STILL HAVE ITS PROCESS ID ENTRY IN THE PROCESS TABLE.

HOW WILL YOU COPY A FILE FROM ONE MACHINE TO OTHER?
WE CAN USE UTILITIES LIKE "FTP," "SCP" OR "RSYNC" TO COPY A FILE FROM ONE MACHINE TO OTHER. E.G., USING FTP:

```
FTP HOSTNAME

>PUT FILE1

>BYE
```

ABOVE COPIES, FILE FILE1 FROM THE LOCAL SYSTEM TO DESTINATION SYSTEM WHOSE HOSTNAME IS SPECIFIED.

I WANT TO MONITOR A CONTINUOUSLY UPDATING LOG FILE, WHAT COMMAND CAN BE USED TO MOST EFFICIENTLY ACHIEVE THIS?
WE CAN USE TAIL –F FILENAME. THIS WILL CAUSE ONLY THE DEFAULT LAST 10 LINES TO BE DISPLAYED ON STD O/P WHICH CONTINUOUSLY SHOWS THE UPDATING PART OF THE FILE.

I WANT TO CONNECT TO A REMOTE SERVER AND EXECUTE SOME COMMANDS, HOW CAN I ACHIEVE THIS?
WE CAN USE SSH TO DO THIS:

```
SSH USERNAME@SERVERIP -P SSHPORT
```

EXAMPLE: SSH ROOT@122.52.251.171 -P 22

ONCE ABOVE COMMAND IS EXECUTED, YOU WILL BE ASKED TO ENTER THE PASSWORD

I HAVE 2 FILES AND I WANT TO PRINT THE RECORDS WHICH ARE COMMON TO BOTH.**
WE CAN USE "COMM" COMMAND AS FOLLOWS:

COMM -12 FILE1 FILE2 ... 12 WILL SUPPRESS THE CONTENT WHICH ARE

UNIQUE TO 1ST AND 2ND FILE RESPECTIVELY.

WRITE A SCRIPT TO PRINT THE FIRST 10 ELEMENTS OF FIBONACCI SERIES.

```
#!/BIN/SH
A=1
B=1
ECHO $A
ECHO $B
FOR I IN 1 2 3 4 5 6 7 8
DO
C=A
B=$A
B=$(($A+$C))
ECHO $B
DONE
```

HOW WILL YOU CONNECT TO A DATABASE SERVER FROM LINUX?
WE CAN USE ISQL UTILITY THAT COMES WITH OPEN CLIENT DRIVER AS FOLLOWS:

```
ISQL –S SERVERNAME –U USERNAME –P PASSWORD
```

WHAT ARE THE 3 STANDARD STREAMS IN LINUX?
0 - STANDARD INPUT1 - STANDARD OUTPUT2 - STANDARD ERROR

I WANT TO READ ALL INPUT TO THE COMMAND FROM FILE1 DIRECT ALL OUTPUT TO FILE2 AND ERROR TO FILE 3, HOW CAN I ACHIEVE THIS?

```
COMMAND <FILE1 1>FILE2 2>FILE3
```

WHAT WILL HAPPEN TO MY CURRENT PROCESS WHEN I EXECUTE A COMMAND USING EXEC?
"EXEC" OVERLAYS THE NEWLY FORKED PROCESS ON THE CURRENT PROCESS; SO WHEN I EXECUTE THE COMMAND USING EXEC, THE COMMAND GETS EXECUTED ON THE CURRENT SHELL WITHOUT CREATING ANY NEW PROCESSES.

E.G., EXECUTING "EXEC LS" ON COMMAND PROMPT WILL EXECUTE LS AND ONCE LS EXITS, THE PROCESS WILL SHUT DOWN

HOW WILL YOU EMULATE WC –L USING AWK?

```
AWK 'END {PRINT NR} FILENAME'
```

GIVEN A FILE FIND THE COUNT OF LINES CONTAINING THE WORD "ABC".

```
GREP –C "ABC" FILE1
```

WHAT IS THE DIFFERENCE BETWEEN GREP AND EGREP?
EGREP IS EXTENDED GREP THAT SUPPORTS ADDED GREP FEATURES LIKE "+" (1 OR MORE OCCURRENCE OF A PREVIOUS CHARACTER),"?"(0 OR 1 OCCURRENCE OF A PREVIOUS CHARACTER) AND "|" (ALTERNATE MATCHING)

HOW WILL YOU PRINT THE LOGIN NAMES OF ALL USERS ON A SYSTEM?
/ETC/SHADOW FILE HAS ALL THE USERS LISTED.

```
AWK –F ':' '{PRINT $1} /ETC/SHADOW'|UNIQ -U
```

HOW TO SET AN ARRAY IN LINUX?
SYNTAX IN KSH:

```
SET –A ARRAYNAME= (ELEMENT1 ELEMENT2 ….. ELEMENT)
```

IN BASH

```
A=(ELEMENT1 ELEMENT2 ELEMENT3 …. ELEMENTN)
```

WRITE DOWN THE SYNTAX OF "FOR " LOOP
SYNTAX:

```
FOR  ITERATOR IN (ELEMENTS)
DO
EXECUTE COMMANDS
DONE
```

HOW WILL YOU FIND THE TOTAL DISK SPACE USED BY A SPECIFIC USER?
DU -S /HOME/USER1 ....WHERE USER1 IS THE USER FOR WHOM THE TOTAL DISK SPACE NEEDS TO BE FOUND.

WRITE THE SYNTAX FOR "IF" CONDITIONALS IN LINUX?
SYNTAX

```
IF  CONDITION IS SUCCESSFUL
THEN
EXECUTE COMMANDS
ELSE
EXECUTE COMMANDS
FI
```

WHAT IS THE SIGNIFICANCE OF $?
THE COMMAND $? GIVES THE EXIT STATUS OF THE LAST COMMAND THAT WAS EXECUTED.

HOW DO WE DELETE ALL BLANK LINES IN A FILE?

```
SED  '^ [(BACKSLASH)011(BACKSLASH)040]*$/D' FILE1
```

WHERE (BACKSLASH)011 IS AN OCTAL EQUIVALENT OF SPACE AND

(BACKSLASH)040 IS AN OCTAL EQUIVALENT OF THE TAB

HOW WILL I INSERT A LINE "ABCDEF" AT EVERY 100TH LINE OF A FILE?

```
SED '100I\ABCDEF' FILE1
```

WRITE A COMMAND SEQUENCE TO FIND ALL THE FILES MODIFIED IN LESS THAN 2 DAYS AND PRINT THE RECORD COUNT OF EACH.**

```
FIND . –MTIME -2 –EXEC WC –L {} ;
```

HOW CAN I SET THE DEFAULT RWX PERMISSION TO ALL USERS ON EVERY FILE WHICH IS CREATED IN THE CURRENT SHELL?
WE CAN USE:

```
UMASK 777
```

THIS WILL SET DEFAULT RWX PERMISSION FOR EVERY FILE WHICH IS CREATED FOR EVERY USER.

HOW CAN WE FIND THE PROCESS NAME FROM ITS PROCESS ID?
WE CAN USE "PS –P PROCESSID"

WHAT ARE THE FOUR FUNDAMENTAL COMPONENTS OF EVERY FILE SYSTEM ON LINUX?
BOOTBLOCK, SUPER BLOCK, INODE BLOCK AND DATABLOCK ARE FOUND FUNDAMENTAL COMPONENTS OF EVERY FILE SYSTEM ON LINUX.

WHAT IS A BOOT BLOCK?
THIS BLOCK CONTAINS A SMALL PROGRAM CALLED "MASTER BOOT RECORD"(MBR) WHICH LOADS THE KERNEL DURING SYSTEM BOOT UP.

WHAT IS A SUPER BLOCK?
SUPER BLOCK CONTAINS ALL THE INFORMATION ABOUT THE FILE SYSTEM LIKE THE SIZE OF FILE SYSTEM, BLOCK SIZE USED BY ITS NUMBER OF FREE DATA BLOCKS AND LIST OF FREE INODES AND DATA BLOCKS.

WHAT IS AN INODE BLOCK?
THIS BLOCK CONTAINS THE INODE FOR EVERY FILE OF THE FILE SYSTEM ALONG WITH ALL THE FILE ATTRIBUTES EXCEPT ITS NAME.

HOW CAN I SEND A MAIL WITH A COMPRESSED FILE AS AN ATTACHMENT?**

```
IP FILE1.ZIP FILE1|MAILX –S "SUBJECT" RECIPIENTS EMAIL ID
```

MAIL CONTENT

OF

. ###### HOW DO WE CREATE COMMAND ALIASES IN A SHELL?

```
ALIAS ALIASNAME="COMMAND WHOSE ALIAS IS TO BE CREATED".
```

WHAT ARE "C" AND "B" PERMISSION FIELDS OF A FILE?
"C " AND "B" PERMISSION FIELDS ARE GENERALLY ASSOCIATED WITH A DEVICE FILE. IT SPECIFIES WHETHER A FILE IS A SPECIAL CHARACTER FILE OR A BLOCK SPECIAL FILE.

WHAT IS THE USE OF A SHEBANG LINE?
SHEBANG LINE AT THE TOP OF EACH SCRIPT DETERMINES THE LOCATION OF THE ENGINE WHICH IS TO BE USED TO EXECUTE THE SCRIPT.

IMPORTANT MISCELLANEOUS
COMMON TCP/IP PROTOCOLS AND PORTS

PROTOCOL	TCP/UDP	PORT NUMBER	DESCRIPTION
FILE TRANSFER PROTOCOL (FTP) (RFC 959)	TCP	20/21	FTP IS ONE OF THE MOST

COMMONLY USED FILE TRANSFER PROTOCOLS ON THE INTERNET AND WITHIN PRIVATE NETWORKS. AN FTP SERVER CAN EASILY BE SET UP WITH LITTLE NETWORKING KNOWLEDGE AND PROVIDES THE ABILITY TO EASILY RELOCATE FILES FROM ONE SYSTEM TO ANOTHER. FTP CONTROL IS HANDLED ON TCP PORT 21 AND ITS DATA TRANSFER CAN USE TCP PORT 20 AS WELL AS DYNAMIC PORTS DEPENDING ON THE SPECIFIC CONFIGURATION.

SECURE SHELL (SSH) (RFC 4250-4256) TCP 22 SSH IS THE PRIMARY METHOD USED TO MANAGE NETWORK DEVICES SECURELY AT THE COMMAND LEVEL. IT IS TYPICALLY USED AS A SECURE ALTERNATIVE TO TELNET WHICH DOES NOT SUPPORT SECURE CONNECTIONS.

TELNET (RFC 854) TCP 23 TELNET IS THE PRIMARY METHOD USED TO MANAGE NETWORK DEVICES AT THE COMMAND LEVEL. UNLIKE SSH WHICH PROVIDES A SECURE CONNECTION, TELNET DOES NOT, IT SIMPLY PROVIDES A BASIC UNSECURED CONNECTION. MANY LOWER LEVEL NETWORK DEVICES SUPPORT TELNET AND NOT SSH AS IT REQUIRED SOME ADDITIONAL PROCESSING. CAUTION SHOULD BE USED WHEN CONNECTING TO A DEVICE USING TELNET OVER A PUBLIC NETWORK AS THE LOGIN CREDENTIALS WILL BE TRANSMITTED IN THE CLEAR.

SIMPLE MAIL TRANSFER PROTOCOL (SMTP) (RFC 5321) TCP 25 SMTP IS USED FOR TWO PRIMARY FUNCTIONS, IT IS USED TO TRANSFER MAIL (EMAIL) FROM SOURCE TO DESTINATION BETWEEN MAIL SERVERS AND IT IS USED BY END USERS TO SEND EMAIL TO A MAIL SYSTEM.

DOMAIN NAME SYSTEM (DNS) (RFC 1034-1035) TCP/UDP 53 THE DNS IS USED WIDELY ON THE PUBLIC INTERNET AND ON PRIVATE NETWORKS TO TRANSLATE DOMAIN NAMES INTO IP ADDRESSES, TYPICALLY FOR NETWORK ROUTING. DNS IS HIERATICAL WITH MAIN ROOT SERVERS THAT CONTAIN DATABASES THAT LIST THE MANAGERS OF HIGH LEVEL TOP LEVEL DOMAINS (TLD) (SUCH AS .COM). THESE DIFFERENT TLD MANAGERS THEN CONTAIN INFORMATION FOR THE SECOND LEVEL DOMAINS THAT ARE TYPICALLY USED BY INDIVIDUAL USERS (FOR EXAMPLE, CISCO.COM). A DNS SERVER CAN ALSO BE SET UP WITHIN A PRIVATE NETWORK TO PRIVATE NAMING SERVICES BETWEEN THE HOSTS OF THE INTERNAL NETWORK WITHOUT BEING PART OF THE GLOBAL SYSTEM.

DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP) (RFC 2131) UDP 67/68 DHCP IS USED ON NETWORKS THAT DO NOT USE STATIC IP ADDRESS ASSIGNMENT (ALMOST ALL OF THEM). A DHCP SERVER CAN BE SET UP BY AN ADMINISTRATOR OR ENGINEER WITH A POLL OF ADDRESSES THAT ARE AVAILABLE FOR ASSIGNMENT. WHEN A CLIENT DEVICE IS TURNED ON IT CAN REQUEST AN IP ADDRESS FROM THE LOCAL DHCP SERVER, IF THERE IS AN AVAILABLE ADDRESS IN THE POOL IT CAN BE ASSIGNED TO THE DEVICE. THIS ASSIGNMENT IS NOT PERMANENT AND EXPIRES AT A CONFIGURABLE INTERVAL; IF AN ADDRESS RENEWAL IS NOT REQUESTED AND THE LEASE EXPIRES THE ADDRESS WILL BE PUT BACK INTO THE POLL FOR ASSIGNMENT.

TRIVIAL FILE TRANSFER PROTOCOL (TFTP) (RFC 1350) UDP 69 TFTP OFFERS A METHOD OF FILE TRANSFER WITHOUT THE SESSION ESTABLISHMENT REQUIREMENTS THAT FTP USES. BECAUSE TFTP USES UDP INSTEAD OF TCP IT HAS NO WAY OF ENSURING THE FILE HAS BEEN PROPERLY TRANSFERRED, THE END DEVICE MUST BE ABLE TO CHECK THE FILE TO ENSURE PROPER TRANSFER. TFTP IS TYPICALLY USED BY DEVICES TO UPGRADE SOFTWARE AND FIRMWARE; THIS INCLUDES CISCO AND OTHER NETWORK VENDORS’ EQUIPMENT.

HYPERTEXT TRANSFER PROTOCOL (HTTP) (RFC 2616) TCP 80 HTTP IS ONE OF THE MOST COMMONLY USED PROTOCOLS ON MOST NETWORKS. HTTP IS THE MAIN PROTOCOL THAT IS USED BY WEB BROWSERS AND IS THUS USED BY ANY CLIENT THAT USES FILES LOCATED ON THESE SERVERS.

POST OFFICE PROTOCOL (POP) VERSION 3 (RFC 1939) TCP 110 POP VERSION 3 IS ONE OF THE TWO MAIN PROTOCOLS USED TO RETRIEVE MAIL FROM A SERVER. POP WAS DESIGNED TO BE VERY SIMPLE BY ALLOWING A CLIENT TO RETRIEVE THE COMPLETE CONTENTS OF A SERVER MAILBOX AND THEN DELETING THE CONTENTS FROM THE SERVER.

NETWORK TIME PROTOCOL (NTP) (RFC 5905) UDP 123 ONE OF THE MOST OVERLOOKED PROTOCOLS IS NTP. NTP IS USED TO SYNCHRONIZE THE DEVICES ON THE INTERNET. EVEN MOST MODERN OPERATING SYSTEMS SUPPORT NTP AS A BASIS FOR KEEPING AN ACCURATE CLOCK. THE USE OF NTP IS VITAL ON NETWORKING SYSTEMS AS IT PROVIDES AN ABILITY TO EASILY INTERRELATE TROUBLES FROM ONE DEVICE TO ANOTHER AS THE CLOCKS ARE PRECISELY ACCURATE.

NETBIOS (RFC 1001-1002) TCP/UDP 137/138/139 NETBIOS ITSELF IS NOT A PROTOCOL BUT IS TYPICALLY USED IN COMBINATION WITH IP WITH THE NETBIOS OVER TCP/IP (NBT) PROTOCOL. NBT HAS LONG BEEN THE CENTRAL PROTOCOL USED TO INTERCONNECT MICROSOFT WINDOWS MACHINES.

INTERNET MESSAGE ACCESS PROTOCOL (IMAP) (RFC 3501) TCP 143 IMAP VERSION3 IS

THE SECOND OF THE MAIN PROTOCOLS USED TO RETRIEVE MAIL FROM A SERVER. WHILE POP HAS WIDER SUPPORT, IMAP SUPPORTS A WIDER ARRAY OF REMOTE MAILBOX OPERATIONS WHICH CAN BE HELPFUL TO USERS.

SIMPLE NETWORK MANAGEMENT PROTOCOL (SNMP) (RFC 1901-1908, 3411-3418)   TCP/UDP   161/162   SNMP IS USED BY NETWORK ADMINISTRATORS AS A METHOD OF NETWORK MANAGEMENT. SNMP HAS A NUMBER OF DIFFERENT ABILITIES INCLUDING THE ABILITY TO MONITOR, CONFIGURE AND CONTROL NETWORK DEVICES. SNMP TRAPS CAN ALSO BE CONFIGURED ON NETWORK DEVICES TO NOTIFY A CENTRAL SERVER WHEN SPECIFIC ACTIONS ARE OCCURRING. TYPICALLY, THESE ARE CONFIGURED TO BE USED WHEN AN ALERTING CONDITION IS HAPPENING. IN THIS SITUATION, THE DEVICE WILL SEND A TRAP TO NETWORK MANAGEMENT STATING THAT AN EVENT HAS OCCURRED AND THAT THE DEVICE SHOULD BE LOOKED AT FURTHER FOR A SOURCE TO THE EVENT.

BORDER GATEWAY PROTOCOL (BGP) (RFC 4271) TCP   179   BGP VERSION 4 IS WIDELY USED ON THE PUBLIC INTERNET AND BY INTERNET SERVICE PROVIDERS (ISP) TO MAINTAIN VERY LARGE ROUTING TABLES AND TRAFFIC PROCESSING. BGP IS ONE OF THE FEW PROTOCOLS THAT HAVE BEEN DESIGNED TO DEAL WITH THE ASTRONOMICALLY LARGE ROUTING TABLES THAT MUST EXIST ON THE PUBLIC INTERNET.

LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL (LDAP) (RFC 4510)   TCP/UDP   389   LDAP PROVIDES A MECHANISM OF ACCESSING AND MAINTAINING DISTRIBUTED DIRECTORY INFORMATION. LDAP IS BASED ON THE ITU-T X.500 STANDARD BUT HAS BEEN SIMPLIFIED AND ALTERED TO WORK OVER TCP/IP NETWORKS.

HYPERTEXT TRANSFER PROTOCOL OVER SSL/TLS (HTTPS) (RFC 2818)TCP   443   HTTPS IS USED IN CONJUNCTION WITH HTTP TO PROVIDE THE SAME SERVICES BUT DOING IT USING A SECURE CONNECTION WHICH IS PROVIDED BY EITHER SSL OR TLS.

LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL OVER TLS/SSL (LDAPS) (RFC 4513) TCP/UDP   636   JUST LIKE HTTPS, LDAPS PROVIDES THE SAME FUNCTION AS LDAP BUT OVER A SECURE CONNECTION WHICH IS PROVIDED BY EITHER SSL OR TLS.

FTP OVER TLS/SSL (RFC 4217)   TCP   989/990   AGAIN, JUST LIKE THE PREVIOUS TWO ENTRIES, FTP OVER TLS/SSL USES THE FTP PROTOCOL WHICH IS THEN SECURED USING EITHER SSL OR TLS.

BASH CHEAT SHEET

EXAMPLE

```
#!/USR/BIN/ENV BASH

NAME="JOHN"
ECHO "HELLO $NAME!"
```

VARIABLES

```
NAME="JOHN"
ECHO $NAME
ECHO "$NAME"
ECHO "${NAME}!"
```

STRING QUOTES

```
NAME="JOHN"
ECHO "HI $NAME"  #=> HI JOHN
ECHO 'HI $NAME'  #=> HI $NAME
```

SHELL EXECUTION

```
ECHO "I'M IN $(PWD)"
ECHO "I'M IN `PWD`"
# SAME
```

CONDITIONAL EXECUTION

```
GIT COMMIT && GIT PUSH
GIT COMMIT || ECHO "COMMIT FAILED"
```

FUNCTIONS

```
GET_NAME() {
  ECHO "JOHN"
}

ECHO "YOU ARE $(GET_NAME)"
```

CONDITIONALS

```
IF [[ -Z "$STRING" ]]; THEN
  ECHO "STRING IS EMPTY"
ELIF [[ -N "$STRING" ]]; THEN
  ECHO "STRING IS NOT EMPTY"
FI
```

STRICT MODE

```
SET -EUO PIPEFAIL
IFS=$'\N\T'
```

BRACE EXPANSION

```
ECHO {A,B}.JS
{A,B} SAME AS A B
{A,B}.JS    SAME AS A.JS B.JS
{1..5}      SAME AS 1 2 3 4 5
```

PARAMETER EXPANSIONS

BASICS

```
NAME="JOHN"
ECHO ${NAME}
ECHO ${NAME/J/J}    #=> "JOHN" (SUBSTITUTION)
ECHO ${NAME:0:2}    #=> "JO" (SLICING)
ECHO ${NAME::2}     #=> "JO" (SLICING)
ECHO ${NAME::-1}    #=> "JOH" (SLICING)
ECHO ${NAME:(-1)}   #=> "N" (SLICING FROM RIGHT)
ECHO ${NAME:(-2):1} #=> "H" (SLICING FROM RIGHT)
ECHO ${FOOD:-CAKE}  #=> $FOOD OR "CAKE"
LENGTH=2
ECHO ${NAME:0:LENGTH}  #=> "JO"
STR="/PATH/TO/FOO.CPP"
ECHO ${STR%.CPP}    # /PATH/TO/FOO
ECHO ${STR%.CPP}.O  # /PATH/TO/FOO.O

ECHO ${STR##*.}     # CPP (EXTENSION)
ECHO ${STR##*/}     # FOO.CPP (BASEPATH)

ECHO ${STR#*/}      # PATH/TO/FOO.CPP
ECHO ${STR##*/}     # FOO.CPP

ECHO ${STR/FOO/BAR} # /PATH/TO/BAR.CPP
STR="HELLO WORLD"
ECHO ${STR:6:5}   # "WORLD"
ECHO ${STR:-5:5}  # "WORLD"
SRC="/PATH/TO/FOO.CPP"
BASE=${SRC##*/}   #=> "FOO.CPP" (BASEPATH)
DIR=${SRC%$BASE}  #=> "/PATH/TO/" (DIRPATH)
```

SUBSTITUTION

```
${FOO%SUFFIX}     REMOVE SUFFIX
${FOO#PREFIX}     REMOVE PREFIX
${FOO%%SUFFIX}    REMOVE LONG SUFFIX
${FOO##PREFIX}    REMOVE LONG PREFIX
${FOO/FROM/TO}    REPLACE FIRST MATCH
```

```
${FOO//FROM/TO}   REPLACE ALL
${FOO/%FROM/TO}   REPLACE SUFFIX
${FOO/#FROM/TO}   REPLACE PREFIX
```

COMMENTS

```
# SINGLE LINE COMMENT
: '
THIS IS A
MULTI LINE
COMMENT
'
```

SUBSTRINGS

```
${FOO:0:3}  SUBSTRING (POSITION, LENGTH)
${FOO:-3:3} SUBSTRING FROM THE RIGHT
```

LENGTH

```
${#FOO}     LENGTH OF $FOO
```

DEFAULT VALUES

```
${FOO:-VAL} $FOO, OR VAL IF NOT SET
${FOO:=VAL} SET $FOO TO VAL IF NOT SET
${FOO:+VAL} VAL IF $FOO IS SET
${FOO:?MESSAGE}   SHOW ERROR MESSAGE AND EXIT IF $FOO IS NOT SET
```

THE : IS OPTIONAL (EG, ${FOO=WORD} WORKS)

LOOPS

BASIC FOR LOOP

```
FOR I IN /ETC/RC.*; DO
  ECHO $I
DONE
```

C-LIKE FOR LOOP

```
FOR ((I = 0 ; I < 100 ; I++)); DO
  ECHO $I
DONE
```

RANGES

```
FOR I IN {1..5}; DO
    ECHO "WELCOME $I"
DONE
```

WITH STEP SIZE

```
FOR I IN {5..50..5}; DO
    ECHO "WELCOME $I"
DONE
```

READING LINES

```
< FILE.TXT | WHILE READ LINE; DO
  ECHO $LINE
DONE
```

FOREVER

```
WHILE TRUE; DO
  ...
DONE
```

FUNCTIONS

DEFINING FUNCTIONS

```
MYFUNC() {
    ECHO "HELLO $1"
}
# SAME AS ABOVE (ALTERNATE SYNTAX)
FUNCTION MYFUNC() {
    ECHO "HELLO $1"
}
MYFUNC "JOHN"
```

RETURNING VALUES

```
MYFUNC() {
    LOCAL MYRESULT='SOME VALUE'
    ECHO $MYRESULT
}
RESULT="$(MYFUNC)"
```

RAISING ERRORS

```
MYFUNC() {
  RETURN 1
}
IF MYFUNC; THEN
  ECHO "SUCCESS"
ELSE
  ECHO "FAILURE"
FI
```

ARGUMENTS

```
$#    NUMBER OF ARGUMENTS
$*    ALL ARGUMENTS
$@    ALL ARGUMENTS, STARTING FROM FIRST
$1    FIRST ARGUMENT
```

CONDITIONALS

CONDITIONS

NOTE THAT [[IS ACTUALLY A COMMAND/PROGRAM THAT RETURNS EITHER 0 (TRUE) OR 1 (FALSE). ANY PROGRAM THAT OBEYS THE SAME LOGIC (LIKE ALL BASE UTILS, SUCH AS GREP(1) OR PING(1)) CAN BE USED AS CONDITION, SEE EXAMPLES.

```
[[ -Z STRING ]]   EMPTY STRING
[[ -N STRING ]]   NOT EMPTY STRING
[[ STRING == STRING ]]  EQUAL
[[ STRING != STRING ]]  NOT EQUAL
[[ NUM -EQ NUM ]] EQUAL
[[ NUM -NE NUM ]] NOT EQUAL
[[ NUM -LT NUM ]] LESS THAN
[[ NUM -LE NUM ]] LESS THAN OR EQUAL
[[ NUM -GT NUM ]] GREATER THAN
[[ NUM -GE NUM ]] GREATER THAN OR EQUAL
[[ STRING =~ STRING ]]  REGEXP
(( NUM < NUM ))   NUMERIC CONDITIONS
[[ -O NOCLOBBER ]]      IF OPTIONNAME IS ENABLED
[[ ! EXPR ]]      NOT
[[ X ]] && [[ Y ]]      AND
`[[ X ]]
```

FILE CONDITIONS

```
[[ -E FILE ]]     EXISTS
[[ -R FILE ]]     READABLE
[[ -H FILE ]]     SYMLINK
[[ -D FILE ]]     DIRECTORY
[[ -W FILE ]]     WRITABLE
[[ -S FILE ]]     SIZE IS > 0 BYTES
[[ -F FILE ]]     FILE
[[ -X FILE ]]     EXECUTABLE
[[ FILE1 -NT FILE2 ]]   1 IS MORE RECENT THAN 2
[[ FILE1 -OT FILE2 ]]   2 IS MORE RECENT THAN 1
[[ FILE1 -EF FILE2 ]]   SAME FILES
```

EXAMPLE

```
IF PING -C 1 GOOGLE.COM; THEN
  ECHO "IT APPEARS YOU HAVE A WORKING INTERNET CONNECTION"
FI
IF GREP -Q 'FOO' ~/.BASH_HISTORY; THEN
  ECHO "YOU APPEAR TO HAVE TYPED 'FOO' IN THE PAST"
FI
# STRING
IF [[ -Z "$STRING" ]]; THEN
  ECHO "STRING IS EMPTY"
ELIF [[ -N "$STRING" ]]; THEN
  ECHO "STRING IS NOT EMPTY"
FI
# COMBINATIONS
IF [[ X ]] && [[ Y ]]; THEN
  ...
FI
# EQUAL
IF [[ "$A" == "$B" ]]
# REGEX
IF [[ "A" =~ "." ]]
IF (( $A < $B )); THEN
   ECHO "$A IS SMALLER THAN $B"
FI
IF [[ -E "FILE.TXT" ]]; THEN
  ECHO "FILE EXISTS"
FI
```

ARRAYS

DEFINING ARRAYS

```
FRUITS=('APPLE' 'BANANA' 'ORANGE')
FRUITS[0]="APPLE"
FRUITS[1]="BANANA"
FRUITS[2]="ORANGE"
```

WORKING WITH ARRAYS

```
ECHO ${FRUITS[0]}           # ELEMENT #0
ECHO ${FRUITS[@]}           # ALL ELEMENTS, SPACE-SEPARATED
ECHO ${#FRUITS[@]}          # NUMBER OF ELEMENTS
ECHO ${#FRUITS}             # STRING LENGTH OF THE 1ST ELEMENT
ECHO ${#FRUITS[3]}          # STRING LENGTH OF THE NTH ELEMENT
ECHO ${FRUITS[@]:3:2}       # RANGE (FROM POSITION 3, LENGTH 2)
```

OPERATIONS

```
FRUITS=("${FRUITS[@]}" "WATERMELON")    # PUSH
FRUITS+=('WATERMELON')                  # ALSO PUSH
FRUITS=( ${FRUITS[@]/AP*/} )            # REMOVE BY REGEX MATCH
UNSET FRUITS[2]                         # REMOVE ONE ITEM
FRUITS=("${FRUITS[@]}")                 # DUPLICATE
```

```
FRUITS=("${FRUITS[@]}" "${VEGGIES[@]}") # CONCATENATE
LINES=(`CAT "LOGFILE"`)                 # READ FROM FILE
```

ITERATION

```
FOR I IN "${ARRAYNAME[@]}"; DO
  ECHO $I
DONE
```

DICTIONARIES

DEFINING

```
DECLARE -A SOUNDS
SOUNDS[DOG]="BARK"
SOUNDS[COW]="MOO"
SOUNDS[BIRD]="TWEET"
SOUNDS[WOLF]="HOWL"
```

DECLARES SOUND AS A DICTIONARY OBJECT (AKA ASSOCIATIVE ARRAY).

WORKING WITH DICTIONARIES

```
ECHO ${SOUNDS[DOG]} # DOG'S SOUND
ECHO ${SOUNDS[@]}   # ALL VALUES
ECHO ${!SOUNDS[@]}  # ALL KEYS
ECHO ${#SOUNDS[@]}  # NUMBER OF ELEMENTS
UNSET SOUNDS[DOG]   # DELETE DOG
```

ITERATION

ITERATE OVER VALUES

```
FOR VAL IN "${SOUNDS[@]}"; DO
  ECHO $VAL
DONE
```

ITERATE OVER KEYS

```
FOR KEY IN "${!SOUNDS[@]}"; DO
  ECHO $KEY
DONE
```

OPTIONS

OPTIONS

```
SET -O NOCLOBBER  # AVOID OVERLAY FILES (ECHO "HI" > FOO)
SET -O ERREXIT    # USED TO EXIT UPON ERROR, AVOIDING CASCADING ERRORS
SET -O PIPEFAIL   # UNVEILS HIDDEN FAILURES
SET -O NOUNSET    # EXPOSES UNSET VARIABLES
```

GLOB OPTIONS

```
SET -O NULLGLOB    # NON-MATCHING GLOBS ARE REMOVED  ('*.FOO' => '')
SET -O FAILGLOB    # NON-MATCHING GLOBS THROW ERRORS
SET -O NOCASEGLOB  # CASE INSENSITIVE GLOBS
SET -O GLOBDOTS    # WILDCARDS MATCH DOTFILES ("*.SH" => ".FOO.SH")
SET -O GLOBSTAR    # ALLOW ** FOR RECURSIVE MATCHES ('LIB/**/*.RB' =>
'LIB/A/B/C.RB')
```

SET GLOBIGNORE AS A COLON-SEPARATED LIST OF PATTERNS TO BE REMOVED FROM GLOB MATCHES.

HISTORY

COMMANDS

```
HISTORY     SHOW HISTORY
SHOPT -S HISTVERIFY     DON’T EXECUTE EXPANDED RESULT IMMEDIATELY
```

EXPANSIONS

```
!$    EXPAND LAST PARAMETER OF MOST RECENT COMMAND
!*    EXPAND ALL PARAMETERS OF MOST RECENT COMMAND
!-N   EXPAND NTH MOST RECENT COMMAND
!N    EXPAND NTH COMMAND IN HISTORY
!<COMMAND>  EXPAND MOST RECENT INVOCATION OF COMMAND <COMMAND>
```

OPERATIONS

```
!!    EXECUTE LAST COMMAND AGAIN
!!:S/<FROM>/<TO>/ REPLACE FIRST OCCURRENCE OF <FROM> TO <TO> IN MOST RECENT COMMAND
!!:GS/<FROM>/<TO>/      REPLACE ALL OCCURRENCES OF <FROM> TO <TO> IN MOST RECENT
COMMAND
!$:T  EXPAND ONLY BASENAME FROM LAST PARAMETER OF MOST RECENT COMMAND
!$:H  EXPAND ONLY DIRECTORY FROM LAST PARAMETER OF MOST RECENT COMMAND
```

!! AND !$ CAN BE REPLACED WITH ANY VALID EXPANSION.

SLICES

```
!!:N  EXPAND ONLY NTH TOKEN FROM MOST RECENT COMMAND (COMMAND IS 0; FIRST ARGUMENT
IS 1)
!^    EXPAND FIRST ARGUMENT FROM MOST RECENT COMMAND
!$    EXPAND LAST TOKEN FROM MOST RECENT COMMAND
!!:N-M      EXPAND RANGE OF TOKENS FROM MOST RECENT COMMAND
!!:N-$      EXPAND NTH TOKEN TO LAST FROM MOST RECENT COMMAND
```

!! CAN BE REPLACED WITH ANY VALID EXPANSION I.E. !CAT, !-2, !42, ETC.

MISCELLANEOUS

NUMERIC CALCULATIONS

```
$((A + 200))      # ADD 200 TO $A
$((RANDOM%=200))  # RANDOM NUMBER 0..200
```

SUBSHELLS

```
(CD SOMEDIR; ECHO "I'M NOW IN $PWD")
PWD # STILL IN FIRST DIRECTORY
```

REDIRECTION

```
PYTHON HELLO.PY > OUTPUT.TXT   # STDOUT TO (FILE)
PYTHON HELLO.PY >> OUTPUT.TXT  # STDOUT TO (FILE), APPEND
PYTHON HELLO.PY 2> ERROR.LOG   # STDERR TO (FILE)
PYTHON HELLO.PY 2>&1           # STDERR TO STDOUT
PYTHON HELLO.PY 2>/DEV/NULL    # STDERR TO (NULL)
PYTHON HELLO.PY &>/DEV/NULL    # STDOUT AND STDERR TO (NULL)
PYTHON HELLO.PY < FOO.TXT      # FEED FOO.TXT TO STDIN FOR PYTHON
```

INSPECTING COMMANDS

```
COMMAND -V CD
#=> "CD IS A FUNCTION/ALIAS/WHATEVER"
```

TRAP ERRORS

```
TRAP 'ECHO ERROR AT ABOUT $LINENO' ERR
```

OR

```
TRAPERR() {
  ECHO "ERROR: ${BASH_SOURCE[1]} AT ABOUT ${BASH_LINENO[0]}"
}

SET -O ERRTRACE
TRAP TRAPERR ERR
```

CASE/SWITCH

```
CASE "$1" IN
  START | UP)
    VAGRANT UP
    ;;

  *)
    ECHO "USAGE: $0 {START|STOP|SSH}"
    ;;
ESAC
```

SOURCE RELATIVE

```
SOURCE "${0%/*}/../SHARE/FOO.SH"
```

PRINTF

```
PRINTF "HELLO %S, I'M %S" SVEN OLGA
#=> "HELLO SVEN, I'M OLGA
```

DIRECTORY OF SCRIPT

```
DIR="${0%/*}"
```

GETTING OPTIONS

```
WHILE [[ "$1" =~ ^- && ! "$1" == "--" ]]; DO CASE $1 IN
  -V | --VERSION )
    ECHO $VERSION
    EXIT
    ;;
  -S | --STRING )
    SHIFT; STRING=$1
    ;;
  -F | --FLAG )
    FLAG=1
    ;;
ESAC; SHIFT; DONE
IF [[ "$1" == '--' ]]; THEN SHIFT; FI
```

HEREDOC

```
CAT <<END
HELLO WORLD
END
```

READING INPUT

```
ECHO -N "PROCEED? [Y/N]: "
READ ANS
ECHO $ANS
READ -N 1 ANS    # JUST ONE CHARACTER
```

SPECIAL VARIABLES

```
$?    EXIT STATUS OF LAST TASK
$!    PID OF LAST BACKGROUND TASK
$$    PID OF SHELL
```

GO TO PREVIOUS DIRECTORY

```
PWD # /HOME/USER/FOO
CD BAR/
PWD # /HOME/USER/FOO/BAR
CD -
PWD # /HOME/USER/FOO
```

3. LINUX COMMANDS:

FILE COMMANDS

COMMAND	DESCRIPTION
LS	DIRECTORY LISTING
LS -AL	FORMATED LISTING WITH HIDDEN FILES
CD DIR	CHANGE DIRECTORY TO DIR
CD	CHANGE TO HOME
PWD	SHOW CURRENT DIRECTORY
MKDIR DIR	CREATE DIRECTORY DIR
RM FILE	DELETE FILE
RM -R DIR	DELETE DIRECTORY DIR
RM -F FILE	FORCE REMOVE FILE
RM -RF DIR	FORCE REMOVE DIRECTORY DIR
CP FILE1 FILE2	COPY FILE1 TO FILE2
CP -R DIR1 DIR2	COPY DIR1 TO DIR2; CREATE DIR2 IF IT DOESN'T EXIST
MV FILE1 FILE2	RENAME OR MOVE FILE1 TO FILE2 IF FILE IS AN EXISTING DIRECTORY, MOVES FILE1 TO SIRECTORY FILE2
LN -S FILE LINK	CREATE SYMBOLIC LINK TO FILE
TOUCH FILE	CREATE OR UPDATE FILE
CAT > FILE	PLACES STANDARD INPUT INTO FILE
MORE FILE	OUTPUT THE CONTENTS OF FILE
HEAD FILE	OUTPUT THE FIRST 10 LINES OF FILE
TAIL FILE	OUTPUT THE LAST 10 LINES OF FILE
TAIL -F FILE	OUTPUT THE CONTENTS OF FILE AS IT GROWS STARTING WITH THE LAST 10 LINES

SYSTEM INFO

COMMAND	DESCRIPTION
DATE	SHOW THE CURRENT DATE AND TIME
CAL	SHOW THIS MONTH'S CALENDAR
UPTIME	SHOW CURRENT UPTIME
W	DISPLAY WHO IS ONLINE
WHOAMI	WHO YOU ARE LOGGED IN AS
FINGER USER	DISPLAY INFO ABOUT USER
UNAME -A	SHOW KERNEL INFO
CAT /PROC/CPUINFO	CPU INFO
CAT /PROC/MEMINFO	MEMORY INFO
MAN COMMAND	SHOW THE MANUAL FOR COMMAND
DF	SHOW DISK USAGE
DU	SHOW DIRECTORY SPACE USAGE
FREE	SHOW MEMORY AND SWAP USAGE
WHEREIS APP	SHOW POSSIBLE LOCATIONS OF APP
WHICH APP	SHOW WHICH APP WILL BE RUN BY DEFAULT

FILE PERMISSIONS COMMANDS

COMMAND	DESCRIPTION
CHMOD OCTAL FILE	CHANGE THE PERMISSIONS OF FILE TO OCTAL, WHICH CAN BE FOUND SEPARATELY FOR USER, GROUP, AND WORLD BY ADDING: 4 – READ (R), 2- WRITE(W), EXECUTE (X)
CHMOD 777	READ, WRITE EXECUTE FOR ALL
CHMOD 755	RWX FOR OWNER, RX FOR GROUP AND WORLD

PROCESS MANAGEMENT COMMANDS

COMMAND	DESCRIPTION
PS	DISPLAY CURRENTLY ACTIVE PROCESSES
TOP	DISPLAY ALL RUNNING PROCESSES
KILL PID	KILL PROCESS ID PID
KILLALL PROC	KILL AL PROCESSES NAMED PROC

```
BG    LISTS STOPPED OR BACKGROUND JOBS; RESUME A STOPPED JOB IN THE BACKGROUND
FG    BRINGS THE MOST RECENT JOB TO THE FOREGROUND
FG N  BRINGS JOB N TO THE FOREGROUND
SSH COMMANDS

COMMAND     DESCRIPTION
SSH USER@HOST     CONNECT TO HOST AS USER
SSH -P PORT USER@HOST   CONNECT TO HOST ON PORT AS USER
SSH-COPY-ID USER@HOST   ADD YOUR KEY TO HOST FOR USER TO ENABLE A KEYED
PASSWORDLESS LOGIN
SEARCHING COMMANDS

COMMAND     DESCRIPTION
GREP PATTERN FILES      SEARCH FOR PATTERN IN FILES
GREO -R PATTERN DIR     SEARCH RECURSIVELY FOR PATTERN IN DIR
COMMAND | GREP PATTERN  SEARCH FOR PATTERN IN THE OUTPUT OF COMMAND
LOCATE FILE FIND ALL INSTANCES OF FILE
COMPRESSION COMMANDS:

COMMAND     DESCRIPTION
TAR CF FILE.TAR FILES   CREATED A TAR NAMED FILE.TAR CONTAINING FILES
TAR XF FILE.TAR   EXTRACT THE FILES FROM FILE.TAR
TAR CZF FILE.TAR.GZ FILES     CREATE A TAR WITH GZIP COMPRESSION
TAR XZF FILE.TAR.GZ     EXTRACT A TAR USING GZIP
TAR CJF FILE.TAR.BZ2    CREATE A TAR WITH BZIP2 COMPRESSION
TAR XJF FILE.TAR.BZ2    EXTRACT A TAR USING BZIP2
GZIP FILE   COMPRESSES FILE AND RENAMES IT TO FILE.GZ
GZIP -D FILE.GZ   DECOMPRESSES FILE.GZ BACK TO FILE
NETWORKING COMMANDS

COMMAND     DESCRIPTION
PING HOST   PING HOST AND OUTPUT RESULTS
WHOIS DOMAIN      GET WHOIS INFORMATION FOR DOMAIN
DIG DOMAIN  GET DNS INFORMATION FOR DOMAIN
DIG -X HOST REVERSE LOOKUP HOST
WGET FILE   DOWNLOAD FILE
WGET -C FILE      CONTINUE A STOPPED DOWNLOAD
NMAP SCAN TYPEOPTIONSTARGET.  SCAN A HOST
IFCONFIG
TRACEROUTE DOMAIN/IP    TRACEROUTE PRINTS THE ROUTE PACKETS TAKE TO NETWORK HOST.
TELNET HOST TALK TO “HOSTS” AT THE GIVEN PORT NUMBER. BY DEFAULT, THE TELNET PORT
IS PORT 23.
NETSTAT –R  PRINT ROUTING TABLES.
ROUTE ADD   USED FOR SETTING A STATIC (NON-DYNAMIC BY HAND ROUTE) ROUTE PATH IN THE
ROUTE TABLES
NSLOOKUP DOMAIN   MAKES QUERIES TO THE DNS SERVER TO TRANSLATE IP TO A NAME, OR
VICE VERSA.
INSTALLATION COMMANDS

COMMAND     DESCRIPTION
MAKE
./CONFIGURE
MAKE INSTALL      INSTALL FROM SOURCE
DPKG -I PKG.DEB   INSTALL A PACKAGE (DEBIAN)
RPM -UVH PKG.RPM  INSTALL A PACKAGE(RPM)
SHORTCUTS

COMMAND     DESCRIPTION
CTRL+C      HALTS THE CURRENT COMMAND
```

CTRL+Z   STOPS THE CURRENT COMMAND, RESUME WITH FG IN THE FOREGROUND OR BG IN THE BACKGROUND

CTRL+D   LOG OUT OF CURRENT SESSION, SIMILAR TO EXIT

CTRL+W   ERASES ONE WORD IN THE CURRENT LINE

CTRL+U   ERASES THE WHOLE LINE

CTRL+R   TYPE TO BRING UP A RECENT COMMAND

!!   REPEATS THE LAST COMMAND

EXIT LOG OUT OF CURRENT SESSION