Process operation of Semaphore

TPYES:
1:) **Binary semaphore**=A semaphore value changes in 01and 10 during the operation called binary semaphore.

2:)**Counting Semaphore**=A semaphore whose token count value is greater than 1 is called as counting semaphore.

Pthread and spinlock

Decration = pthread_spinlock_t my_spin;
init = pthread_spinlock_init(pthread_spinlock_t *ptr

Context Switching = switching off cpu in one executing process to another task

Deadlock Situation = Deadlock is a situation is nither the of this process will move further.Each process is waiting for other process to release the lock.

Mutex =
1:) Mutex are similar to semaphore with a usage count of one (binary semaphore)

atomic_t is a data type declear atomic variable .Atomic are vey tiny locks,int variable whenever decelere a varible type atomic_t type kernal provide a set of function to safe guard the declere integer variable

initialize atomic_t u = ATOMIC_INIT(0);

atomic_t is ansure safty of intiger varible on con-current axis

pthread Mutex

pthread detached

Decleration Pthread_one_t = isto schedule and execute initialization code of type which take to NO argument NO return

initialize is pthread_one_t onces;

pthread library provide a macro.

pthread_once(pthread_once_t **ptr, void (***function pointer)(void));

Pthread Schedular Attribute

1. I**nherit schedular member** = Thats spacifies us whether a thread is inheriting policy and priority from parentd and managing by its own.

a. Pthread_inherit_sched
b. Pthread_explicit_sched

2.Scheduling polity & priority=

When a thread create a new thread a newly created thread will inherit schedular properties of parents

if the policy is default all the thread will have all tha cpu have same time sharing

LINUX employees complete(CFS)

each and every process will get proportion of CPU based on factor called as Nice Value.
Nice Value gives waitage to the process with request to cpu . Used for priority of process

if nice value is same for all processes each every process will get a timeslace of =1/n* cpu time.

Each and Every process are going to get equal proportion to get equal giving time slice next process

---

----------16/03/2022----------

PROCESS CONTEXT

KERNAL CONTEXT

INTERRUPT CONTEXT

At any givin of time kernal will be executed either in process context ,kernal context or interrupt context

Kernal executed a piece of software on behalf of process which as initiated a system call is called as process context.

Kenal context = Kernal executrd a pieces of software(kernal service) on behalf of another kernal service

Each and every on your system architechture connect to pic controller through a physical wire called as interrupt request line

Kernal executing a piece of software on behalf on interrup is called as interrupt context

LINUX MEMORY MANAGEMENT UNIT

Memory manipulation calls

void *memset(void * addr,int c,size_t n bytes);

when process execute memset function jumb to the address location provided by pointer argument and start set the data with given constant (c) for n number of bytes.

The moment process execute memchar function jump to location provided by the pointer variable and starts scanning giving constant C.

The moment process execute memcmp function it jump to two address location and start compareing address bytes by bytes until it get unmatching data and return +1,-1 and 0;

the moment process execute memmove operation copy the data from source buffer to destination buffer for given N no. Of bytes.

Memmove	memcpy
slow	fast
reliable & generate	not reliable
in case of memory overlaping source and destination memmove provide	there is no temperary buffer

use tempory buffer

the moment process execute ALLOCA memory call alocates memory from stack segment(heap segment is full) and return pointer to the allocate region on success

alloca(size_t n byte

Minimun memory wastage and operation is faster campare to malloc operation .If repected called heap is full so stack overflow occur.

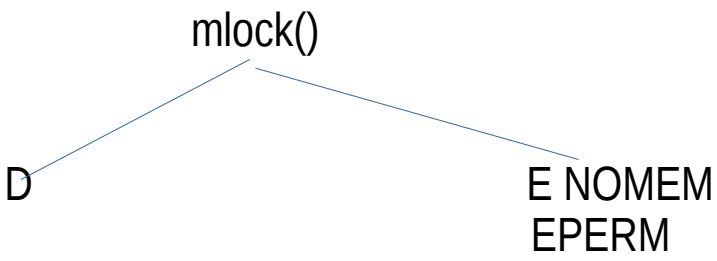
alloc dosent mention pool of fixed block sizes and enhance no memory fragmentation didnt need to call free memory is automatically delocated bcoz memo terminates when function ends.

----------SWAPING PROCESS----------

when ram running out of memory and os want to launch new application into the ram then os looks inactive for process into ram and push them into swap partition into storege divices\(harddisk) this is called swapout process

when newly application done their job kernam will get back the processes from swap partition to ram and this is called as swapin process.

Memory locks there would be a login program and we dont want kernal to swapout my login process from the ram then we can apply memory lock



EAGAIN

ENOMEM = the error return bu mlock when tryin to applying more then permitted limit

EPERM =when no privillage return a error is EPERM when it is fail to apply memory lock for spacified address

sbrk and brk are used to manage memory data segment shrk takes a value by which create a new program breakpoint and brk function take desired address to generate a new program break point.

----------17/03/22----------

MMAP OPERATION

Problem with user space or an kernal space :-
If an application making repeatitive io request a much cpu time spend in submitted io operation

mmap is fosics memory function that map that a givin kernal file reagion or a divice region or some kernal memory into the process at a space

mmap syntax & argument:-

mmap(void * address, size_t bytes,   int protection, int flag, fd/-1, off_t offset);

void *address== where is my

size_t = no. Of bytes we want to map.

Protection=1. PROT_READ (PAGES MAY BE READ)
2. PROT_WRITE (PAGES MAY BE WRITE)
3. PROT_EXEC (PAGES MAY BE EXECUTE)

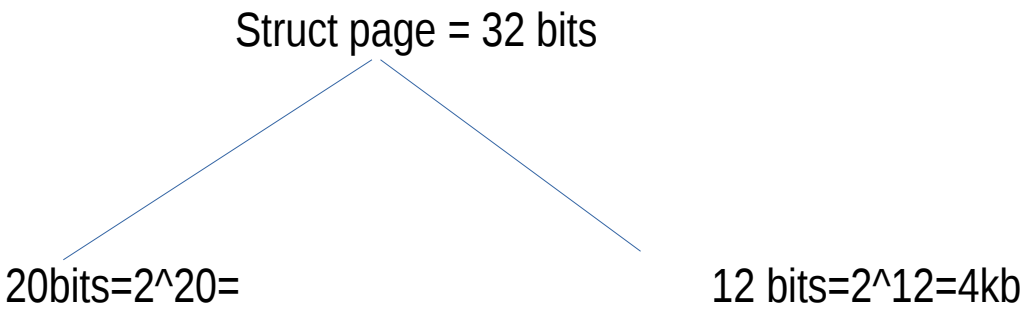
FLAG = MAP_ANONYMOUS (does not belong to any file)
MAP_LOCKED (applying to mlock to sharred region)
MAP_SHARRED (sharred b/w n no. of region)
MAP_PRIVATE (create a private copy and changes will not reflect to other process which are mapping to sharred region)

address augument 0 = It is recommanded to use zero indicate asking to kernal to map free process at a space

fd = -1 when file not exsit

when kernal booting start low level MMU will also starts and create & initialize lot of kernal memory data structure.

low level mmu convert all memory into struct pages.



Each and every process maintain one page table entry inside the PCB of process.

If reference to the PTE is lost karnal will never know where the process pages are in kernal memory.
PTE identify this particular pages belong to X particular process.

Inside the kernal there is a dynamic data structure called as page frame relation table.

Now processor will take the page no. and for look for a matching frame no. in page frame relation table fatches the respected page no. And than takes the page number and add to offset leading to the pysical address of varible x

For each and every io request opertaion processor have to perform logical to physical address translation.

VIRTUAL MEMORY

virtual memory exception of memory created tempararyly in swap partion of storage devices
Linux uses ram and the virtual memory and virtual address have assign to process.

>free- command provide amount of available virtual memory
> v command is very similar to vmstat -S
>cat proc/meminfo
>vmstat- virtual memory statistic reportor, I
>free- free available virtual memory
>bi – block received virtual memory
>bo- block send to block code
>in – no. Of interrupt per second
>cs – number of context per second

LIBRARY is a group of pre-compile object code.

STATIC library are one which are statically link to program executable file at compile time

DYNAMIC library are which are dyanmically link to program executable file at run time

linux static library has a extension is .a

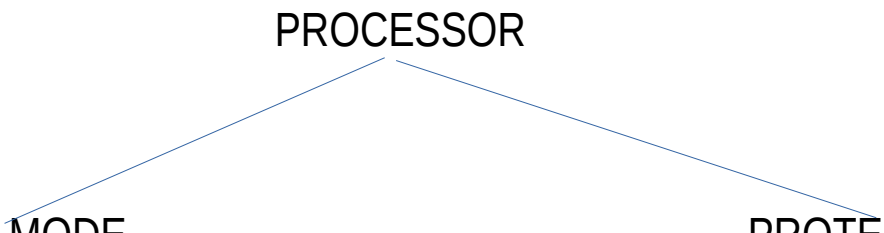
----------21/03/22----------

Dynamic file has a exterension is .so and also called as shades library

Dynamic library = creation uses a flag called as PIC (Process independent code) so dynamic library relocatable that means instruction of dynamic library should get ready to get load into any memory at a space .EX- Executable file need dynamic library Process 1 need the address of dynamic library and dynamic should merge with memory executable program in such a way dynamic library should apper as a part of execulatbe program later if another executalbe file required a dynamically library the instrution of library should merge with memory of the execulatble program.

-shared =

A tool LDD is called as linux command line tool will provide the dependance required for the executable files



Its all depend upon processor how its looking at memorybased on which OS will foam policy to manage memory processor will look into memory in 2 different

once your system power on during bias code or bootloader code processor in real mode and looking for memory as array on bytes

Once the kernal bootup start processor shifted to protected mode and looking memory as a set of blocks (linked list of pages) and this process called process initialization.

kernal OS creating some illusion and making CPU to look memory as set of block

DEBUGGING GDB(DEBUGGER

is a open source execulatble file by compilation process

its a powerful tool that support many languages c,c++,java,forton etc

BREAK_POINT gdb use break command to stop the program execution at our line of interest

If he absorb real time debug opertion most of the times bugs happened to be from user point of view EX- accidentally dereperencing the null pointer, dereperencing the intialized pointer, dereperencing the unintialized pointer , acessing the memory beyond lower boundary region

when a process created process address at a space allocated and program expected is to use only the allocated region whenever a program make a violation or whenever a program make memory violation then segmentation error occur.
CPU will get logicall addresses and will map to repective physicall addresses with the help of kernal frame relation table in case logical address are invaild memory addresses cpu giving invaild address stored and kernel fails to mao get respected frame number for invaild addresses

processor through page fold error to the kernel and kernal will check the process segmented area and then send a signal called as SIGSEGY

signal to the repected process on the reception of signal the process will abrupt the operation a segmentation fault

INFOLOCALS will dipslay value o all variables.

----------22/03/2022----------

gdb call variable p type

attribute align of structure spacified plz provide minimum alignment for the structure variable

GDB failed to track down heap memory allocation

electric-fan is an library not an debugging tool which has got its own implementation of malloc and calloc memory
using electic-fan when requesting dinamic memory allocations using malloc and calloc will aloocate for spacified amount of memory

Standard c library malloc and calloc allocating more no. Of bytes then actually requested for

electric -fan configure to report heap memory vilalations either for uper boundary region or upper boundary region but not both at a time

Cofigurinf electric -fans to report lower boundary region

VALGRIND = is a runtime tool use to trap down heap memory violation is also called as heap memory profiling tool is a stand alone debugger used at runtime.When process make use of standard c library malloc and calloc calls valgrind reports why this process is failed

Use valgring tool use with and without debugger
x buff printing ascii value at present in particular
x/s buff will print your string until u reach

instead of re-compling the code change can be make from gdb command to modifird the value of exsistance variable will help of gdb command called set

----------23/03/2022----------

MAKE FILE

Makefile is called as program building tools in linux and unix operating system .Makefile is a set of command similar to terminal commands but diiference is make file have an organised text instruction .Makefile will have which called variabe.

*Kernal build makefile ,
*Entire Linux kernel OS,
*Is build into a makefile called as kernel build make file located at this location=cd */lib*/modules/5.04.0-104-generic/build/

*process architecture makefile

cd */lib*modules/<kernal version>/build/arch/x86

Rules to write Makefile
Makefile should start with upper case M without any extension all the source files and makefile should be in current directory

Makefile should start with target collen
next instruction should start with

make is a linux utilities tool use to generate application execute make will reduce work load on compilation

Make tool will execute the makefile of current directory and jumps to makefile and start exectution of makefile gcc is indirectly called by makefile intern develops application

The primary object of make tool is to break down large codes source code into small pieces and access small piece of code weather required recompilation or not

Makefile contain make varible and may contain single target or multiple target is a file name that is generated by

A target is an apllication executable or an target is an object file or tagey may be action such as clean

appln: main.o mean.o sub.o -o appln is an target dependency
here we have variable and multiple target here create object file as a target .If i want to re-complie any individual file this make will reduces workload of re-compile all file bcoz make is going to re-compile only the modified file.

system(Fork ,Excel , Wait)
#include<stdlib.h>
int system(const char* command);

error = child not crated
error = too little arguments

then system calls an empty

the retrun system system call Zero and one values will talk about status of shell when you are getting zero shell is not abvailable,NULL provide as a command the retuen value is One that is non zero value,its indicate shell will available

----------24/03/2022----------

pthread_exit will terminate the current task and allows the pending task to get execute

Pthread condition variable w.r.t mutex

```
pthread_cond_t   mycond;
```

pthread_cond_wait(pthread_cond_t *ptr,pthread_mutex_t *mt)

pthread is aquiring a mutex law an wait for event to happen then called pthread condition wait call the moment process execute pthread condition thread call thread is unlocking mutex lock automatically and places itself in wait queue of condition variable .

SIGNAL IS TO WALKUP ONLY ONE THREAD

pthread condition = pthread condition signal function will read the condition varibale argument if any thread waiting on the condition will send a backup signal to the thread

Pthread condition broadcast function = If a thread execute pthread broadcast condition function reads for the condition argument and backup all the threads which are sleeping on condition variable.

Linux threads will have default scheduling policy sched_other,sched_normal in this case the priority is dynamic

Default scheduling policy priority is dynamic that can we change by the dynamic system behaviour of the thread

Sched_FIFO
schedulling policy thread will have fixed priority 1 to 99 ...1 is lowest and 99 is highest

A thread with priority one in sched FIFO policy will execute first when compare to default policy thread

round robin schedulling policy = each and every threrad will have equal priority and will execute in circular order .each thread will uses all the resource of the same amount of time

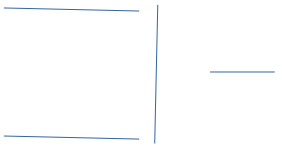
pthread_getschedparam (pthread_self(),&policy,&param);

----------25/03/2022----------

Inter Process Communication

HTTP client process
Emails,SMS
Each machine of the network is identify by the a unique 32 bits ip address it used to connect or communication and 16 bit port address is used to identify process in particular nodes
Linux have productifed IPC techniques;

1. PIPES
2. FIFO
3. Message Queue
4. Shared Memory
5. Semophore



IPC 5 standard

1.PIPES

*Serial communication device that permits unidirectional data transfer.

*Can be used b/w Parent and child process(related process).

Create A pipes

1. int a[2];
2. int read_fd;
3. int write_fd;
4. pipe();
5. data to write into pipes

When a pipe is getting created then the a pipe gets created in kernal spaces and pipes also return 2 discriptor.Fd=0 is always associated to read a file and Fd=1 associated to write a file.

When parent process calling fork child is created.

Child inherite a pipe file from parent thus a pipe is limited to parent and child process

chl\ild process read data from readend

#After a write operation parent flushes the data which data immidiately reflected on readend.

LIMITATION of pipe capacity

1. pipe has limited size to set a data.

Pipe capacity is limited

writer process writing in a write end with a higher speed then the recevier process which is consuming the data it is slower rate

After somethime pipes get full if writer process want to write data but not able to write either writer have a data as a result writer process block until somespace or room create the pipe(child is reading).

When space create the pipe is geting unblock and again start write data into the pipes

if pipes is empty reader process getting block. Until some data return onto the pipe.the mpment data return on a pipe reader is geting unlock

mkfifo filename command is used to create a fifo

Types of File

1. Regular file
2. Directory file

3. pipe file(2 fd)
4. character device file
5. block device files
6. socket file

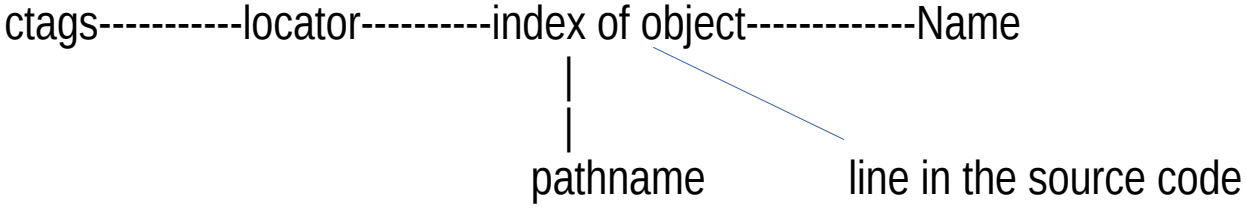
----------28/03/2022----------

Examing C Source Code

Cscope is a linux utilies tools used for software development process to exam C source Code you can exam the the symbols (Variable, Macrod and function) of source code.
To what value their intialize and what program .. using cscope you can verify specific function calling all other function .You can find specific files,You can check for # including header files,You can assign variable option to change the value.

Ctag generate a tag file that is a index file for the names found in source code and header file. Name path name the line where we used in source code . Ctag provide a quick references to your source code example:- Find a defination of particular function

Ctags uses locators and locators will locates the object and path name of the object and line in a source code and out of this it is going to generate tag file quick references

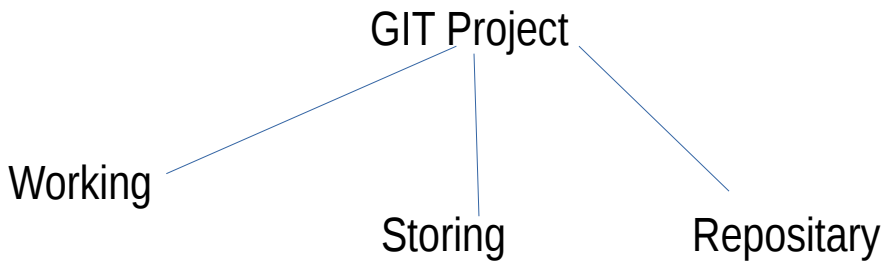


SOURCE CONTROL/ VERSION CONTROL

1. Allow you to tract down your files program of your project
2. Allow you to store all modification and changed given
3. Protecting against date
4. Complex project Development

source control or version control alows you to track down files and files progress for a period of time.

Linux is massive project on github supported by 1000 of contributors



git configuration variable = git config --global username

----------30/03/2022----------

SHARED MEMORY

shared memory is one of the IPC technique shared memory allow 2 or more process to acess a given region of shared memory.shared memory fastest IPC technique bcoz didnot to jump from one memory location to other memory location for data read and write operation b/w reader and writer process and client & server process

shared memory can be used into server and client machine.only trick is synchronize acess of given shared region with semophore technique.Each and every shared memory has a structure called as shmid_ds

int shmget(key_t key, size_t n bytes, int flag);

----------31/03/2022----------

LINUX KERNAL ARCHITECTURE

a device driver is a piece of paper that communicate the hardware and manage the hardware and bring the functionility of the end user to the user

In linux OS device driver reside in the kernal space.

A driver has two interfaces blw application and driver that is OS specific and driver and hardware

Semaphore is an one of the IPC techique deals with array of semaphore.Array of semaphore is bit complex issue but in large application software process need to work on lots of resources need more protection then having array of semophore is big advantage.

Semget() , semop() ,semctl() sem_undo ,is an array of semaphore.

Semop() Is used to change the value of semophore.

semctl() is used to control the semophore operation .Int command is the set value SETVAL command is used to intialize semophote to a val .the value is required to pass value member of uinion SEM_SEMAN only then the processor is perform when we are using semophore for the first time.

Linux OS single = Signals are software interrruoprts they notify process about an event occur signals is asyncronus in nature. A cpu kernal or any software that is running on cpu can trigger signals to the process .A process is an enough permission sent signal to process and . A process can signal to itself.

Terminal Generating Signals

the signal generated by certain terminal key

Linux signal having a naming convection starting which 3 character SIG .Each and every signal defined by a number provided by a header file <signal.h>

Linux classified into 2 categories

Hardware exception signal = invalid memory reference,divided/0

Instead of running default action on delevaring of signal A user can program user define function can end with register with kernal. Own delivary of a signal kernal should invoke kernal define function.

Signal Handler
User define function
signal take two argument

----------04/04/2022----------

SIGNAL HANDLER

Case ||
application X want to perfrom periodic task function then rasister with a timer sub system for delivering of a signal it a perticular time out .

Timer subsystem collect the time slices(timer subsysytem invoke the signal subsystem after some time off and timer subsytem will then delivery a signal called sigalarm(SIGALAM)

I/O signal is delivered by I/O subsystem to the processes when particular file discriptor and socket ready to perform I/O operation.

IN signal subsystem there is a function clled sand_ signal and there fn is executed before delivary of a signal

Steps Follower Sand_Siignal

1. Will get reference of pcb to which signal to be delivered and get point to be signal structure will point to pcb there is signal pointer to signal structure that is pointing to a vactor of 62 elements

2. Sandsignal will manupilate signal structure (for ex sigint,second element of the vector)

A time need to block signal ,a process dealing with a critical section and critical section is creating a database. During this updation process doesnot want to pream then process can block particular process.

----------05/04/2022----------

Mistake in programing code is error if the error detected by a tester is called as a defect .If the defect is accepted by repected developer is called as bug and problem of aloving bug is called debugging.When a system software fails to perform particular function execution leads to failure

FAULT=
Fault is a condition because which a system software is fail

Static Code Analyzer
Is process of identifing programing error and bugs in the source code before the progrm is being run ,SCA is done on set of instruction by using some coding standards .These kind of analysis help to identify a loop hole and the weeknesses and source code that might be harmful.Analysis a stationary peace of spftware therefore its called as SCA

SPLINT is an static code analyzer tool is used to identify the programming error and suspicious suspension and stylistic error
Programming Errors=

Respomsibility of linker

linker job is to provide linker asdd runtime code to build executable ,rumtime is not an library is an set of routine added by the linker during programming build time_start,_init,_fini the moment of your execution start its start with _init is also called as initializer,initializer is also reloacte key locatopn for and also for (that is providing address to an object file for load and execute once address ia an configure then ctrl goes to _start start macro is preretun to execute main fn and ctrl jump to rum main fn execution here the application personlity execute thern when function terminate is also terminated again the control goes to start macro and start macro called _fini macro

clang is an compiler which campare c and c++ and bulid using c++ and release using apache 2.0 licence. Clang is faster and compare to gcc
ex:- google chrome browser for window ,is no bulid using c++ clang compare
clang only support few environment

----------06/04/2022----------

Vender of hte mother board is providing address of bias code and bootloader code .
The kernal image Vmlinux
the firsr piece of code bootcode will execute and shift the cpu to rral mode to protected maode and process is called protect intialization.

Execute until shut down initiated .
Role of kernal
1. Setup Memory DS
2. Interrrupt ds
3. device
4.process
5.file
6. initialize CPU schdular
7.initialize kernal thread

#DYNAMIC TOOL ANALYSIS OF SOURCE CODE

GCOV = GCC COVERAGE TOOL which is an open source tool

gcov is used to analysis of your source code and it will check for untested part of the source code and all identify unexecuted instructions can also we used as profiling tool and browing and navigation of source code allows you to modified enhance the source code

When we are using -fprofile-arcs

#NETWORK PROGRAMMING

N/W operation

there programs communicate program to same machine or pragram on different machine location

LAN works on broad cast approch without having any intermiadiate switch bcoz which data rate in lan much greater then wan

CAN it dedicate to establish between two point between several nodes a path is connection of sequence of physical links b/w the nodes (cable line) if sender want to transmit packet the packets goes to these physical links EXAMPLE telephone communication,packet switch netwok,

In packet switch network that divides the data into small packets .Netwok packet transfer through the network(digital format 0's and 1's) Ex:- wifi

Router ,switch,hub,bridge,

Router and switches are networking devices there used to connecting one or more device for other computer or other netwoking devices or other network.

HUB is used to connected devices in LAN

----------07/04/2022----------

session layer provide connection between the sender and recevier. Session layer stop transmission ,a next cycle session layer didnot start again from begining .Its continues transmitting from point of interruption (synchronization)

#transport layer recevies the data from session layer and divide

Network take the data from transport layer and convert into

network layer has ip address of source and destination ,all the routing process recevier of the packet

Data link layer checking for error ,checking data iswhether data is error or error free and remove the error and transfer error free packet forword

it also maintains the data rate speed sender or recevier,

Traffic may increse on the receving side as a result recevier may loss the packet it is maintein a common dataa rate speeds b/w sender and recevier ,

phsical addressing job
digital format will convert the data int o's and 1's

Aphysical link layer divieding router switches and convert into electrical signal and redio waves .

$ ifconfig
mtu is maximum transmission unit

txqueuelen is deals with

loopback address = each device has loopback address 127.0.0.1 is also called as local host .it used for testing purpose, So when we send the data using loopback address data never resches the network,data is an

loop in a network ,loopback address is used for testing tcp/ip internal flow path
loopback address will help to device to send and receive packet to other

we can say that socket system is design to suppose network communication protocal
bcoz of this same region socket system parameters are generic in nature

As we are using socket system calls will use sam socket structure as argument and take as structure struct sockaddr is also take size of parameter this will identify size of socket structure

socket() creates an endpoint for communication and returns a file descriptor that refers to that endpoint. The file descriptor returned by a successful call will be the lowest-numbered file descriptor not currently open for the process.

Basically need two socket at sender and receiver

socket parameter need 5
1. protocals
2. IP address of source
3. IP address of destination
4. port address of source
5. port address of destination

network application programimg all about application and writing cleint server program.

<sys/types.h>
<sys/socket.h>

struct sockaddr
{
sin-family; -----> N/W protocal,TCP/ip,
sin-port; ----->16 bit port no;(n/w byte order)

```
        sin-addr;          ----->32 bit IP address
}
```

int socket(int family,int type,int protocal);
int bind(int socket,struct sockaddr *saw,int addrlen

AF_INET , SOCK_STREAM,

in ip protocal header the protocal value is zero for ip based operation
on successful echibution of socket file discriptor,socket are nothing

Process operation of Semaphore

TPYES:
1:) **Binary semaphore**=A semaphore value changes in 01and 10 during the operation called binary semaphore.

2:)**Counting Semaphore**=A semaphore whose token count value is greater than 1 is called as counting semaphore.

Pthread and spinlock

Decration = pthread_spinlock_t my_spin;
init = pthread_spinlock_init(pthread_spinlock_t *ptr

Context Switching = switching off cpu in one executing process to another task

Deadlock Situation = Deadlock is a situation is nither the of this process will move further.Each process is waiting for other process to release the lock.

Mutex =
1:) Mutex are similar to semaphore with a usage count of one (binary semaphore)

atomic_t is a data type declear atomic variable .Atomic are vey tiny locks,int variable whenever decelere a varible type atomic_t type kernal provide a set of function to safe guard the declere integer variable

initialize atomic_t u = ATOMIC_INIT(0);

atomic_t is ansure safty of intiger varible on con-current axis

pthread Mutex

pthread detached

Decleration Pthread_one_t = isto schedule and execute initialization code of type which take to NO argument NO return

initialize is pthread_one_t onces;

pthread library provide a macro.

pthread_once(pthread_once_t **ptr, void (***function pointer)(void));

Pthread Schedular Attribute

1. I**nherit schedular member** = Thats spacifies us whether a thread is inheriting policy and priority from parentd and managing by its own.

a. Pthread_inherit_sched
b. Pthread_explicit_sched

2.Scheduling polity & priority=

When a thread create a new thread a newly created thread will inherit schedular properties of parents

if the policy is default all the thread will have all tha cpu have same time sharing

LINUX employees complete(CFS)

each and every process will get proportion of CPU based on factor called as Nice Value.
Nice Value gives waitage to the process with request to cpu . Used for priority of process

if nice value is same for all processes each every process will get a timeslace of =1/n* cpu time.

Each and Every process are going to get equal proportion to get equal giving time slice next process

---

----------16/03/2022----------

PROCESS CONTEXT

KERNAL CONTEXT

INTERRUPT CONTEXT

At any givin of time kernal will be executed either in process context ,kernal context or interrupt context

Kernal executed a piece of software on behalf of process which as initiated a system call is called as process context.

Kenal context = Kernal executrd a pieces of software(kernal service) on behalf of another kernal service

Each and every on your system architechture connect to pic controller through a physical wire called as interrupt request line

Kernal executing a piece of software on behalf on interrup is called as interrupt context

LINUX MEMORY MANAGEMENT UNIT

Memory manipulation calls

void *memset(void * addr,int c,size_t n bytes);

when process execute memset function jumb to the address location provided by pointer argument and start set the data with given constant (c) for n number of bytes.

The moment process execute memchar function jump to location provided by the pointer variable and starts scanning giving constant C.

The moment process execute memcmp function it jump to two address location and start compareing address bytes by bytes until it get unmatching data and return +1,-1 and 0;

the moment process execute memmove operation copy the data from source buffer to destination buffer for given N no. Of bytes.

Memmove	memcpy
slow	fast
reliable & generate	not reliable
in case of memory overlaping source and destination memmove provide	there is no temperary buffer

use tempory buffer

the moment process execute ALLOCA memory call alocates memory from stack segment(heap segment is full) and return pointer to the allocate region on success

alloca(size_t n byte

Minimun memory wastage and operation is faster campare to malloc operation .If repected called heap is full so stack overflow occur.

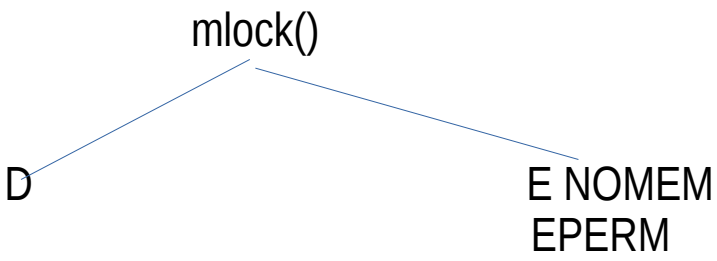
alloc dosent mention pool of fixed block sizes and enhance no memory fragmentation didnt need to call free memory is automatically delocated bcoz memo terminates when function ends.

----------SWAPING PROCESS----------

when ram running out of memory and os want to launch new application into the ram then os looks inactive for process into ram and push them into swap partition into storege divices\(harddisk) this is called swapout process

when newly application done their job kernam will get back the processes from swap partition to ram and this is called as swapin process.

Memory locks there would be a login program and we dont want kernal to swapout my login process from the ram then we can apply memory lock



EAGAIN

ENOMEM = the error return bu mlock when tryin to applying more then permitted limit

EPERM =when no privillage return a error is EPERM when it is fail to apply memory lock for spacified address

sbrk and brk are used to manage memory data segment shrk takes a value by which create a new program breakpoint and brk function take desired address to generate a new program break point.

----------17/03/22----------

MMAP OPERATION

Problem with user space or an kernal space :-
If an application making repeatitive io request a much cpu time spend in submitted io operation

mmap is fosics memory function that map that a givin kernal file reagion or a divice region or some kernal memory into the process at a space

mmap syntax & argument:-

mmap(void * address, size_t bytes,   int protection, int flag, fd/-1, off_t offset);

void *address== where is my

size_t = no. Of bytes we want to map.

Protection=1. PROT_READ (PAGES MAY BE READ)
2. PROT_WRITE (PAGES MAY BE WRITE)
3. PROT_EXEC (PAGES MAY BE EXECUTE)

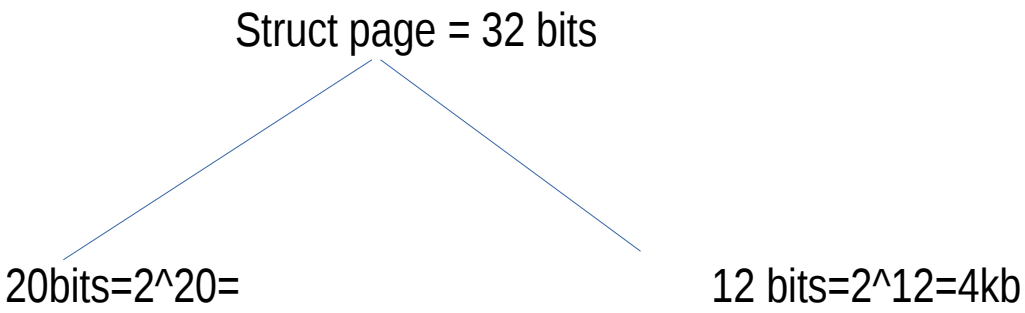
FLAG = MAP_ANONYMOUS (does not belong to any file)
MAP_LOCKED (applying to mlock to sharred region)
MAP_SHARRED (sharred b/w n no. of region)
MAP_PRIVATE (create a private copy and changes will not reflect to other process which are mapping to sharred region)

address augument 0 = It is recommanded to use zero indicate asking to kernal to map free process at a space

fd = -1 when file not exsit

when kernal booting start low level MMU will also starts and create & initialize lot of kernal memory data structure.

low level mmu convert all memory into struct pages.



Each and every process maintain one page table entry inside the PCB of process.

If reference to the PTE is lost karnal will never know where the process pages are in kernal memory.
PTE identify this particular pages belong to X particular process.

Inside the kernal there is a dynamic data structure called as page frame relation table.

Now processor will take the page no. and for look for a matching frame no. in page frame relation table fatches the respected page no. And than takes the page number and add to offset leading to the pysical address of varible x

For each and every io request opertaion processor have to perform logical to physical address translation.

VIRTUAL MEMORY

virtual memory exception of memory created tempararyly in swap partion of storage devices
Linux uses ram and the virtual memory and virtual address have assign to process.

>free- command provide amount of available virtual memory
> v command is very similar to vmstat -S
>cat proc/meminfo
>vmstat- virtual memory statistic reportor, I
>free- free available virtual memory
>bi – block received virtual memory
>bo- block send to block code
>in – no. Of interrupt per second
>cs – number of context per second

LIBRARY is a group of pre-compile object code.

STATIC library are one which are statically link to program executable file at compile time

DYNAMIC library are which are dyanmically link to program executable file at run time

linux static library has a extension is .a

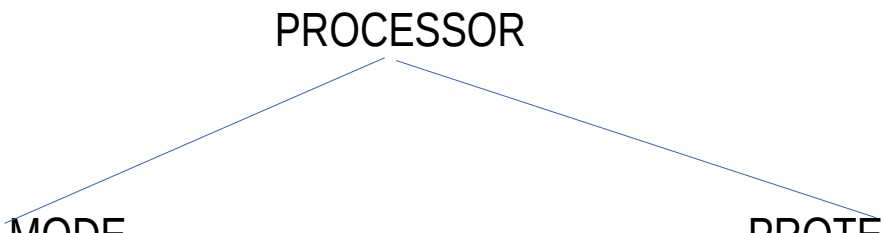
----------21/03/22----------

Dynamic file has a exterension is .so and also called as shades library

Dynamic library = creation uses a flag called as PIC (Process independent code) so dynamic library relocatable that means instruction of dynamic library should get ready to get load into any memory at a space .EX- Executable file need dynamic library Process 1 need the address of dynamic library and dynamic should merge with memory executable program in such a way dynamic library should apper as a part of execulatbe program later if another executalbe file required a dynamically library the instrution of library should merge with memory of the execulatble program.

-shared =

A tool LDD is called as linux command line tool will provide the dependance required for the executable files



Its all depend upon processor how its looking at memorybased on which OS will foam policy to manage memory processor will look into memory in 2 different

once your system power on during bias code or bootloader code processor in real mode and looking for memory as array on bytes

Once the kernal bootup start processor shifted to protected mode and looking memory as a set of blocks (linked list of pages) and this process called process initialization.

kernal OS creating some illusion and making CPU to look memory as set of block

DEBUGGING GDB(DEBUGGER

is a open source execulatble file by compilation process

its a powerful tool that support many languages c,c++,java,forton etc

BREAK_POINT gdb use break command to stop the program execution at our line of interest

If he absorb real time debug opertion most of the times bugs happened to be from user point of view EX- accidentally dereperencing the null pointer, dereperencing the intialized pointer, dereperencing the unintialized pointer , acessing the memory beyond lower boundary region

when a process created process address at a space allocated and program expected is to use only the allocated region whenever a program make a violation or whenever a program make memory violation then segmentation error occur.
CPU will get logicall addresses and will map to repective physicall addresses with the help of kernal frame relation table in case logical address are invaild memory addresses cpu giving invaild address stored and kernel fails to mao get respected frame number for invaild addresses

processor through page fold error to the kernel and kernal will check the process segmented area and then send a signal called as SIGSEGY

signal to the repected process on the reception of signal the process will abrupt the operation a segmentation fault

INFOLOCALS will dipslay value o all variables.

----------22/03/2022----------

gdb call variable p type

attribute align of structure spacified plz provide minimum alignment for the structure variable

GDB failed to track down heap memory allocation

electric-fan is an library not an debugging tool which has got its own implementation of malloc and calloc memory
using electic-fan when requesting dinamic memory allocations using malloc and calloc will aloocate for spacified amount of memory

Standard c library malloc and calloc allocating more no. Of bytes then actually requested for

electric -fan configure to report heap memory vilalations either for uper boundary region or upper boundary region but not both at a time

Cofigurinf electric -fans to report lower boundary region

VALGRIND = is a runtime tool use to trap down heap memory violation is also called as heap memory profiling tool is a stand alone debugger used at runtime.When process make use of standard c library malloc and calloc calls valgrind reports why this process is failed

Use valgring tool use with and without debugger
x buff printing ascii value at present in particular
x/s buff will print your string until u reach

instead of re-compling the code change can be make from gdb command to modifird the value of exsistance variable will help of gdb command called set

----------23/03/2022----------

MAKE FILE

Makefile is called as program building tools in linux and unix operating system .Makefile is a set of command similar to terminal commands but diiference is make file have an organised text instruction .Makefile will have which called variabe.

*Kernal build makefile ,
*Entire Linux kernel OS,
*Is build into a makefile called as kernel build make file located at this location=cd */lib*/modules/5.04.0-104-generic/build/

*process architecture makefile

cd */lib*modules/<kernal version>/build/arch/x86

Rules to write Makefile
Makefile should start with upper case M without any extension all the source files and makefile should be in current directory

Makefile should start with target collen
next instruction should start with

make is a linux utilities tool use to generate application execute make will reduce work load on compilation

Make tool will execute the makefile of current directory and jumps to makefile and start exectution of makefile gcc is indirectly called by makefile intern develops application

The primary object of make tool is to break down large codes source code into small pieces and access small piece of code weather required recompilation or not

Makefile contain make varible and may contain single target or multiple target is a file name that is generated by

A target is an apllication executable or an target is an object file or tagey may be action such as clean

appln: main.o mean.o sub.o -o appln is an target dependency
here we have variable and multiple target here create object file as a target .If i want to re-complie any individual file this make will reduces workload of re-compile all file bcoz make is going to re-compile only the modified file.

system(Fork ,Excel , Wait)
#include<stdlib.h>
int system(const char* command);

error = child not crated
error = too little arguments

then system calls an empty

the retrun system system call Zero and one values will talk about status of shell when you are getting zero shell is not abvailable,NULL provide as a command the retuen value is One that is non zero value,its indicate shell will available

----------24/03/2022----------

pthread_exit will terminate the current task and allows the pending task to get execute

Pthread condition variable w.r.t mutex

```
pthread_cond_t   mycond;
```

pthread_cond_wait(pthread_cond_t *ptr,pthread_mutex_t *mt)

pthread is aquiring a mutex law an wait for event to happen then called pthread condition wait call the moment process execute pthread condition thread call thread is unlocking mutex lock automatically and places itself in wait queue of condition variable .

SIGNAL IS TO WALKUP ONLY ONE THREAD

pthread condition = pthread condition signal function will read the condition varibale argument if any thread waiting on the condition will send a backup signal to the thread

Pthread condition broadcast function = If a thread execute pthread broadcast condition function reads for the condition argument and backup all the threads which are sleeping on condition variable.

Linux threads will have default scheduling policy sched_other,sched_normal in this case the priority is dynamic

Default scheduling policy priority is dynamic that can we change by the dynamic system behaviour of the thread

Sched_FIFO
schedulling policy thread will have fixed priority 1 to 99 ...1 is lowest and 99 is highest

A thread with priority one in sched FIFO policy will execute first when compare to default policy thread

round robin schedulling policy = each and every threrad will have equal priority and will execute in circular order .each thread will uses all the resource of the same amount of time

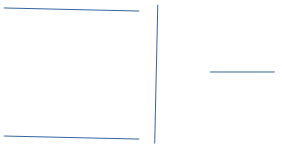
pthread_getschedparam (pthread_self(),&policy,&param);

----------25/03/2022----------

Inter Process Communication

HTTP client process
Emails,SMS
Each machine of the network is identify by the a unique 32 bits ip address it used to connect or communication and 16 bit port address is used to identify process in particular nodes
Linux have productifed IPC techniques;

1. PIPES
2. FIFO
3. Message Queue
4. Shared Memory
5. Semophore



IPC 5 standard

1.PIPES

*Serial communication device that permits unidirectional data transfer.

*Can be used b/w Parent and child process(related process).

Create A pipes

1. int a[2];
2. int read_fd;
3. int write_fd;
4. pipe();
5. data to write into pipes

When a pipe is getting created then the a pipe gets created in kernal spaces and pipes also return 2 discriptor.Fd=0 is always associated to read a file and Fd=1 associated to write a file.

When parent process calling fork child is created.

Child inherite a pipe file from parent thus a pipe is limited to parent and child process

chl\ild process read data from readend

#After a write operation parent flushes the data which data immidiately reflected on readend.

LIMITATION of pipe capacity

1. pipe has limited size to set a data.

Pipe capacity is limited

writer process writing in a write end with a higher speed then the recevier process which is consuming the data it is slower rate

After somethime pipes get full if writer process want to write data but not able to write either writer have a data as a result writer process block until somespace or room create the pipe(child is reading).

When space create the pipe is geting unblock and again start write data into the pipes

if pipes is empty reader process getting block. Until some data return onto the pipe.the mpment data return on a pipe reader is geting unlock

mkfifo filename command is used to create a fifo

Types of File

1. Regular file
2. Directory file

3. pipe file(2 fd)
4. character device file
5. block device files
6. socket file

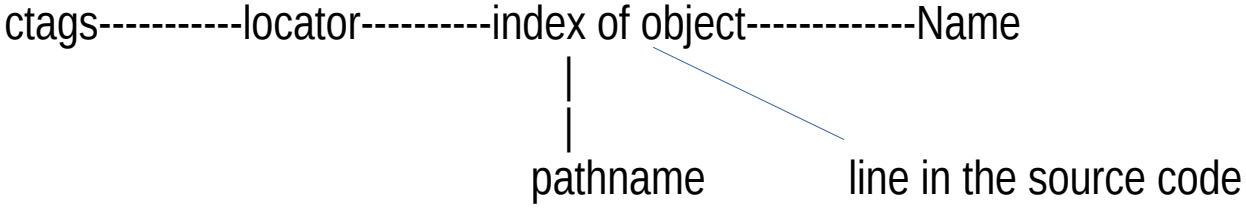
----------28/03/2022----------

Examing C Source Code

Cscope is a linux utilies tools used for software development process to exam C source Code you can exam the the symbols (Variable, Macrod and function) of source code.
To what value their intialize and what program .. using cscope you can verify specific function calling all other function .You can find specific files,You can check for # including header files,You can assign variable option to change the value.

Ctag generate a tag file that is a index file for the names found in source code and header file. Name path name the line where we used in source code . Ctag provide a quick references to your source code example:- Find a defination of particular function

Ctags uses locators and locators will locates the object and path name of the object and line in a source code and out of this it is going to generate tag file quick references

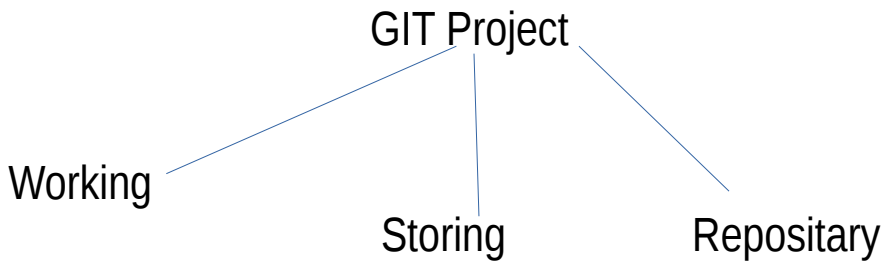


SOURCE CONTROL/ VERSION CONTROL

1. Allow you to tract down your files program of your project
2. Allow you to store all modification and changed given
3. Protecting against date
4. Complex project Development

source control or version control alows you to track down files and files progress for a period of time.

Linux is massive project on github supported by 1000 of contributors



git configuration variable = git config --global username

----------30/03/2022----------

SHARED MEMORY

shared memory is one of the IPC technique shared memory allow 2 or more process to acess a given region of shared memory.shared memory fastest IPC technique bcoz didnot to jump from one memory location to other memory location for data read and write operation b/w reader and writer process and client & server process

shared memory can be used into server and client machine.only trick is synchronize acess of given shared region with semophore technique.Each and every shared memory has a structure called as shmid_ds

int shmget(key_t key, size_t n bytes, int flag);

----------31/03/2022----------

LINUX KERNAL ARCHITECTURE

a device driver is a piece of paper that communicate the hardware and manage the hardware and bring the functionility of the end user to the user

In linux OS device driver reside in the kernal space.

A driver has two interfaces blw application and driver that is OS specific and driver and hardware

Semaphore is an one of the IPC techique deals with array of semaphore.Array of semaphore is bit complex issue but in large application software process need to work on lots of resources need more protection then having array of semophore is big advantage.

Semget() , semop() ,semctl() sem_undo ,is an array of semaphore.

Semop() Is used to change the value of semophore.

semctl() is used to control the semophore operation .Int command is the set value SETVAL command is used to intialize semophote to a val .the value is required to pass value member of uinion SEM_SEMAN only then the processor is perform when we are using semophore for the first time.

Linux OS single = Signals are software interrruoprts they notify process about an event occur signals is asyncronus in nature. A cpu kernal or any software that is running on cpu can trigger signals to the process .A process is an enough permission sent signal to process and . A process can signal to itself.

Terminal Generating Signals

the signal generated by certain terminal key

Linux signal having a naming convection starting which 3 character SIG .Each and every signal defined by a number provided by a header file <signal.h>

Linux classified into 2 categories

Hardware exception signal = invalid memory reference,divided/0

Instead of running default action on delevaring of signal A user can program user define function can end with register with kernal. Own delivary of a signal kernal should invoke kernal define function.

Signal Handler
User define function
signal take two argument

----------04/04/2022----------

SIGNAL HANDLER

Case ||
application X want to perfrom periodic task function then rasister with a timer sub system for delivering of a signal it a perticular time out .

Timer subsystem collect the time slices(timer subsysytem invoke the signal subsystem after some time off and timer subsytem will then delivery a signal called sigalarm(SIGALAM)

I/O signal is delivered by I/O subsystem to the processes when particular file discriptor and socket ready to perform I/O operation.

IN signal subsystem there is a function clled sand_ signal and there fn is executed before delivary of a signal

Steps Follower Sand_Siignal

1. Will get reference of pcb to which signal to be delivered and get point to be signal structure will point to pcb there is signal pointer to signal structure that is pointing to a vactor of 62 elements

2. Sandsignal will manupilate signal structure (for ex sigint,second element of the vector)

A time need to block signal ,a process dealing with a critical section and critical section is creating a database. During this updation process doesnot want to pream then process can block particular process.

----------05/04/2022----------

Mistake in programing code is error if the error detected by a tester is called as a defect .If the defect is accepted by repected developer is called as bug and problem of aloving bug is called debugging.When a system software fails to perform particular function execution leads to failure

FAULT=
Fault is a condition because which a system software is fail

Static Code Analyzer
Is process of identifing programing error and bugs in the source code before the progrm is being run ,SCA is done on set of instruction by using some coding standards .These kind of analysis help to identify a loop hole and the weeknesses and source code that might be harmful.Analysis a stationary peace of spftware therefore its called as SCA

SPLINT is an static code analyzer tool is used to identify the programming error and suspicious suspension and stylistic error
Programming Errors=

Respomsibility of linker

linker job is to provide linker asdd runtime code to build executable ,rumtime is not an library is an set of routine added by the linker during programming build time_start,_init,_fini the moment of your execution start its start with _init is also called as initializer,initializer is also reloacte key locatopn for and also for (that is providing address to an object file for load and execute once address ia an configure then ctrl goes to _start start macro is preretun to execute main fn and ctrl jump to rum main fn execution here the application personlity execute thern when function terminate is also terminated again the control goes to start macro and start macro called _fini macro

clang is an compiler which campare c and c++ and bulid using c++ and release using apache 2.0 licence. Clang is faster and compare to gcc
ex:- google chrome browser for window ,is no bulid using c++ clang compare
clang only support few environment

----------06/04/2022----------

Vender of hte mother board is providing address of bias code and bootloader code .
The kernal image Vmlinux
the firsr piece of code bootcode will execute and shift the cpu to rral mode to protected maode and process is called protect intialization.

Execute until shut down initiated .
Role of kernal
1. Setup Memory DS
2. Interrrupt ds
3. device
4.process
5.file
6. initialize CPU schdular
7.initialize kernal thread

#DYNAMIC TOOL ANALYSIS OF SOURCE CODE

GCOV = GCC COVERAGE TOOL which is an open source tool

gcov is used to analysis of your source code and it will check for untested part of the source code and all identify unexecuted instructions can also we used as profiling tool and browing and navigation of source code allows you to modified enhance the source code

When we are using -fprofile-arcs

#NETWORK PROGRAMMING

N/W operation

there programs communicate program to same machine or pragram on different machine location

LAN works on broad cast approch without having any intermiadiate switch bcoz which data rate in lan much greater then wan

CAN it dedicate to establish between two point between several nodes a path is connection of sequence of physical links b/w the nodes (cable line) if sender want to transmit packet the packets goes to these physical links EXAMPLE telephone communication,packet switch netwok,

In packet switch network that divides the data into small packets .Netwok packet transfer through the network(digital format 0's and 1's) Ex:- wifi

Router ,switch,hub,bridge,

Router and switches are networking devices there used to connecting one or more device for other computer or other netwoking devices or other network.

HUB is used to connected devices in LAN

----------07/04/2022----------

session layer provide connection between the sender and recevier. Session layer stop transmission ,a next cycle session layer didnot start again from begining .Its continues transmitting from point of interruption (synchronization)

#transport layer recevies the data from session layer and divide

Network take the data from transport layer and convert into

network layer has ip address of source and destination ,all the routing process recevier of the packet

Data link layer checking for error ,checking data iswhether data is error or error free and remove the error and transfer error free packet forword

it also maintains the data rate speed sender or recevier,

Traffic may increse on the receving side as a result recevier may loss the packet it is maintein a common dataa rate speeds b/w sender and recevier ,

phsical addressing job
digital format will convert the data int o's and 1's

Aphysical link layer divieding router switches and convert into electrical signal and redio waves .

$ ifconfig
mtu is maximum transmission unit

txqueuelen is deals with

loopback address = each device has loopback address 127.0.0.1 is also called as local host .it used for testing purpose, So when we send the data using loopback address data never resches the network,data is an

loop in a network ,loopback address is used for testing tcp/ip internal flow path
loopback address will help to device to send and receive packet to other

we can say that socket system is design to suppose network communication protocal
bcoz of this same region socket system parameters are generic in nature

As we are using socket system calls will use sam socket structure as argument and take as structure struct sockaddr is also take size of parameter this will identify size of socket structure

socket() creates an endpoint for communication and returns a file descriptor that refers to that endpoint. The file descriptor returned by a successful call will be the lowest-numbered file descriptor not currently open for the process.

Basically need two socket at sender and receiver

socket parameter need 5
1. protocals
2. IP address of source
3. IP address of destination
4. port address of source
5. port address of destination

network application programimg all about application and writing cleint server program.

<sys/types.h>
<sys/socket.h>

struct sockaddr
{
sin-family; -----> N/W protocal,TCP/ip,
sin-port; ----->16 bit port no;(n/w byte order)

```
        sin-addr;          ----->32 bit IP address
}
```

int socket(int family,int type,int protocal);
int bind(int socket,struct sockaddr *saw,int addrlen

AF_INET , SOCK_STREAM,

in ip protocal header the protocal value is zero for ip based operation
on successful echibution of socket file discriptor,socket are nothing