

GROUP 72

Members:

Prakriti Jain (2022362)

Varsha Manoj (2021297)

Contribution:

Both of us worked on the assignment together

Implementation:

- In the SmartLoader file, the main() function calls the load_and_run_elf() function with ELF file as an argument
- The load_and_run_elf() function gets the entry point address from the ELF header and type casts it to a function pointer type of _start()
- Since this will generate a segmentation fault, a signal handler my_handler() is created.
- When a segmentation fault occurs, the signal handler will check the address at which the fault occurs and allocate memory for that address. Then, it reads the ELF header and a loop iterates through the program headers to find the segment that corresponds to the address that creates the segmentation fault. The program then copies the segment data.
- The handler then allocates memory for the segment using mmap(). Then, the segment data is read and copied into the allocated memory.
- After the signal handler exits, the program continues execution as the memory can be accessed.
- After execution, it will print the total number of page faults, the total number of page allocations carried out for that executable's execution and the total amount of internal fragmentation in KB.
- The function loader_cleanup() closes the file descriptor and frees the memory allocated.

Link to Github Repository:

https://github.com/varsha21297/OS_Assignment