SimpleSmartLoader in C

Contributors: Aditya Gupta & Abhishek Bansal

Individual Contribution

- Aditya: segfault hander initial implementation and logic , implemented page by page for bonus , confirming the output and making test case to distinguish for bonus .
- Abhishek: updated and finalized the segfault handler , error handling , storing the mmaped segments and munmapped them by using array implementation .

Implementation

the steps for the implementation of the Smartloader are as follows:

1. This assignment is a continuation of the first assignment thus instead of mmap directly and then typecasting the start functional pointer directly to the ehdr.e_entry .
2. This will create a segmentation fault as no physical page is allocated to that virtual address . Now we are using sigaction struct which will be used to call the SIGSEGV signal . This will call the segfault_handler whenever segmentation fault comes .
3. Now we are mapping the required segments by searching the fault_address in the range of the segment ie . phdr[segment_number].p_v_addr and phdr[segment_number].p_v_addr+phdr[segment_number].p_memsz . Thus this is how we get the required segment . Now we will put this segment in the segment in one-shot(without bonus) in this . The arguments of the mmap are as desired and we will just lseek the fd to correct position(new position) and just read this fd in the segment loaded .
4. This is the basic implementation of the handler and after adding all the desired segments segmentation fault will be removed . Now after the segfault is removed , the program will start running from calling the start functional pointer and the program works completeky fine .'
5. Now for the bonus part , we are adding segments of PAGE_SIZE(4096) . Thus we are adding the segments page by page in each segfault whenever the faulty address come the segfault will end and thus the program will end and run smoothly without allocation of additional pages . A test case for the same has been attached .

Github Repository

Steps to run:

With-Bonus

1. Run the Makefile
2. Now run the commands.
3. ./loader_bonus ./fib
4. ./loader_bonus ./sum
5. ./loader_bonus ./prime
6. ./loader_bonus ./sum_better

Without-Bonus

1. Run the Makefile
2. Now run the commands.
3. ./loader ./fib
4. ./loader ./sum
5. ./loader ./prime
6. ./loader ./sum_better