

Jitendra Kumar Khasdev

17-March-2015

Mapping of functions (library or user defined) to system call.

Abstract:

Mapping of functions with respective system calls will the help to give closer view of interaction between kernel space and user space. Prime focus is on to simulate user space function to kernel space function. This project will simulate linking between user space function to kernel space function. Adding this feature to strace, it will make it more powerful for visualization. Strace can do effectively call by call mapping with respective system call.

Content:

Description and why it is innovative

When a function is called into a simple c program, ultimately it invokes a system call to get service by the kernel. Most of the time, user (developer/administrator etc.) having a difficulty to find which function is invoking a particular system call.

Let us assume two different library functions invoking same system call simultaneously. Now, user has to put very hard effort to differentiate which function is linked with a particular system call. Currently strace displays system calls called by a program without any callee function description.

Our project will map each user space function to corresponding system call in a proper view format, so that user will not having trouble to simulate user space function to kernel space function. It will make debugging and tracing fast and more efficient.

Design :

Step 1: First user will fire a binary of program with strace command with specified flag.

Step 2: Whenever strace will detect particular flag then it will perform following steps else it will

give result as it is as it gives.

Step 3: It will collect the list of functions which is used into a particular program from symbol table or may be any other way. [Preliminary focused on symbol table].

Step 4: Now perform mapping of each function to corresponding system call and display into such kind of format [format of display is not fixed it is depends on degree of attractiveness].

eg. Printf() calls internally write() syscall, so output format as,

```
printf ()      -----  
-> | write(1, buffer, sizeof(buffer)); |  
      -----
```

Step 5 : Exit

it will trace function calls like debugger fashion but limit of tracing will be until it reach to a appropriate system call. It will not only give broad view of working of the system but also it will be handy hacking tool as well as it will be helpful for new bodies to apply the concept of computer science.

That will increase strace audience area as well as it will be a more strong and powerful tool.

Education:

Currently pursuing Bachelor's in Engineering from Computer Science and Engineering with 8.02 CGPA.

Higher secondary school 78.6 %

High school 87.2%

Courseware:

C, Data structure, Operating system, Linux, Computer system organisation, Networking, OOT, Principle of programming language, Software engineering and project management etc.

Training:

C, Linux, System programming

Projects:**NEKTech-Linux-shell development:**

It is Unix like a shell for nektch-linux, I have worked two feature of it (cd support, and cd ~ support).

System Identification Module:

It is utility which keep record of active client machines over the server. Basically it based on product activation and deactivation model.

Gesture detection based security system:

In which, it detects predefined gesture of the person, and take action according to that gesture.

Activities:

1. My one Linux kernel patch is accepted by the Linux community, however it was just coding style fix but I become aware of **code review process** of Linux community.
2. I am aware of version controlling software and I have worked **git**.
3. I am good c, Linux and Operating System concept and enthusiastic about system programming and kernel.
4. I have communicated in open source community with developer via mailing list.
5. Solved coding problems on hackerrank.com, and some on spoj.com
6. Participated into eudypota challenge solved till level 6.

I am focused on only this project and period of next 4-5 month I am free, because I am on end session of my engineering so it will be a great chance for me to work for real world before entering into a professional engineering life.