BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY (BiT)

FACUITY OF computing

Operating system and system programming

Individual Assignment

Name: Tsehay Adugna

Section: A

ID Number :1602632

Title:System call

Submission date:16/08/2017

Submitted to:Lec Wendemu B

Pivot_root()

system call is a programmatic way in which a computer program requests a service from the kernel of the operating system it is executed on.

pivot_root() is a system call that changes the root filesystem (i.e., /) of the current process to a new location. It's commonly used in container technologies, initramfs scripts, or custom boot environments.

Prototype:

int pivot_root(const char *new_root, const char *put_old);

•new_root: The new root directory.

•put_old: A location under the new root where the old root will be mounted (so it can later be unmounted if desired).

Example Use Case:

When booting a system using an initramfs, pivot_root is used to switch from the temporary initial root filesystem (like a RAM disk) to the real root filesystem (like the SD card or another partition).

Requirements:

•Both new_root and put_old must be on the same filesystem.

•pivot_root requires root privileges.

•Not all environments (like some containers) allow its usage due to limited capabilities.

Implementation of pivot_root() on Raspberry Pi:

This script assumes you're in a minimal boot environment, Such as an initramfs ora live rescue shell - not from a fully running Raspberry Pi 0S.

#!/bin/sh

 # Mount essential filesystems mount -t proc none /proc mount -t sysfs none /sys mount -t devtmpfs none /dev

 # Mount new root filesystem (replace with actual device) mount

 /dev/sda1 /mnt

Prepare old_root directory

mkdir -p /mnt /old_root # Move into new root cd /mnt # Do the pivot pivot_root . old_root root

Unmount unnecessary mounts from old root umount /old__root/proc umount /old__root/sys umount /old_root/dev

There's a minimal C code example that calls the pivot_root() system call directly.

```
#define _GNU_SOURCE

#include <unistd.h>

#include <stdio.h>

#include <sys/syscall.h>

#include <errno.h>


int main() {

   const char *new_root = "/newroot";

   const char *put_old = "/newroot/old_root";


   int ret = syscall(SYS_pivot_root, new_root, put_old);

   if (ret == 0) {

      printf("pivot_root successful.\n");

   } else {

      perror("pivot_root failed");

   }


   return 0;

}
```