```
/*
 * File: tun-01.c
 * Author: Songqing Hua
 *
 * (C) 2023 Songqing Hua.
 * https://blog.csdn.net/whowin/
 *
 * Set up tun0 device in a simple way under Linux
 * Compiled with gcc version 9.4.0 (Ubuntu 9.4.0-1ubuntu1~20.04.1).
 * Tested on Linux 5.4.0-139-generic #156-Ubuntu SMP Fri Jan 20 17:27:18 UTC 2023 x86_64 x86_64
x86_64 GNU/Linux.
 *
 * To compile: $ gcc -Wall tun-01.c -o tun-01
 * Usage: $ sudo ./tun-01
 *
 * Example source code for article 銆婁娇鐢ㄣun锛氭嫙缃戝崱鍜岄攣鍖呰IP閿ㄢ叺鑶勫憱渚嬪瓙€€
 *
 */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <fcntl.h>
#include <errno.h>
#include <signal.h>

#include <linux/if.h>
#include <linux/if_tun.h>

#include <sys/types.h>
#include <sys/ioctl.h>

#include <arpa/inet.h>

#define BUFSIZE         2048
#define TUN_DEV_NAME    "tun0"
#define TUN_DEV_FILE    "/dev/net/tun"

int tun_fd = 0;
unsigned long int tun_count = 0;

void sigint(int signum);

int main(int argc, char *argv[]) {
    uint16_t nread;
    char buffer[BUFSIZE];
    struct ifreq ifr;

    // Open device file /dev/net/tun
    tun_fd = open(TUN_DEV_FILE , O_RDWR);
    // Register device name into Linux kernel
    memset(&ifr, 0, sizeof(ifr));
    ifr.ifr_flags = IFF_TUN | IFF_NO_PI;
    strcpy(ifr.ifr_name, TUN_DEV_NAME);
    ioctl(tun_fd, TUNSETIFF, (void *)&ifr);
    printf("Successfully connected to interface %s\n", TUN_DEV_NAME);
    signal(SIGINT, sigint);
    // set up IP address for tun0
    system("ifconfig tun0 10.0.0.1/24 up");
    // Process the packet received from tun0
    while (1) {
        nread = read(tun_fd, buffer, BUFSIZE);
        printf("TUN %lu: Read %d bytes from the tun interface\n", ++tun_count, nread);
    }

    if (tun_fd) close(tun_fd);
    return 0;
}
```

```
void sigint(int signum) {
    // Clean up ......
    if (tun_fd > 0) close(tun_fd);
    printf("Terminating....\n");
    printf("Totally packets from tun: %ld packets\n", tun_count);
    exit(EXIT_SUCCESS);
}
```