#----------------------------------------------------------------------------#

# Author: Ark Sun #

# [arksun9481@gmail.com](mailto:arksun9481@gmail.com) #

#----------------------------------------------------------------------------#

Project github: <https://github.com/ark-sun/IoT-Security>

Cloud server setup

My environment:

Ubuntu 18.04 (64bit)

You can choose other Linux version as you want.

Install python3:

Sudo apt-get install python3

Git clone the project:

cd ~/Desktop/

git clone <https://github.com/ark-sun/IoT-Security.git>

Cd IoT-Security/

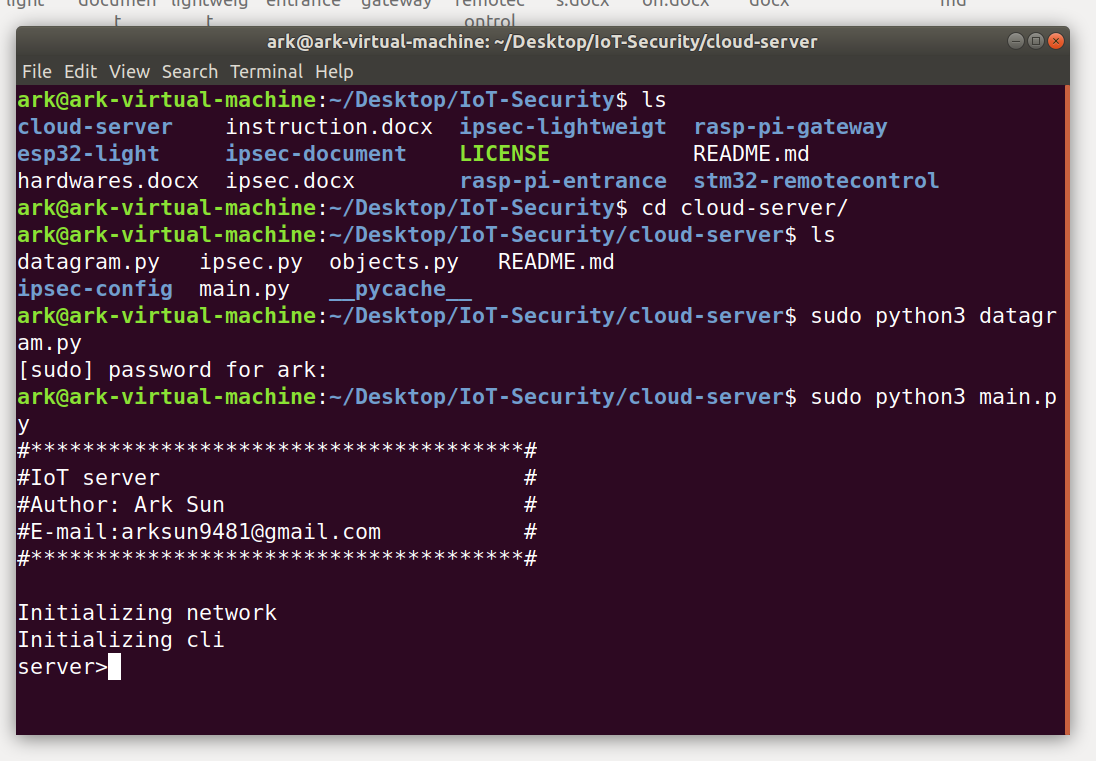
The project folder as following



Cd cloud-server/

Sudo python3 main.py # run the server software

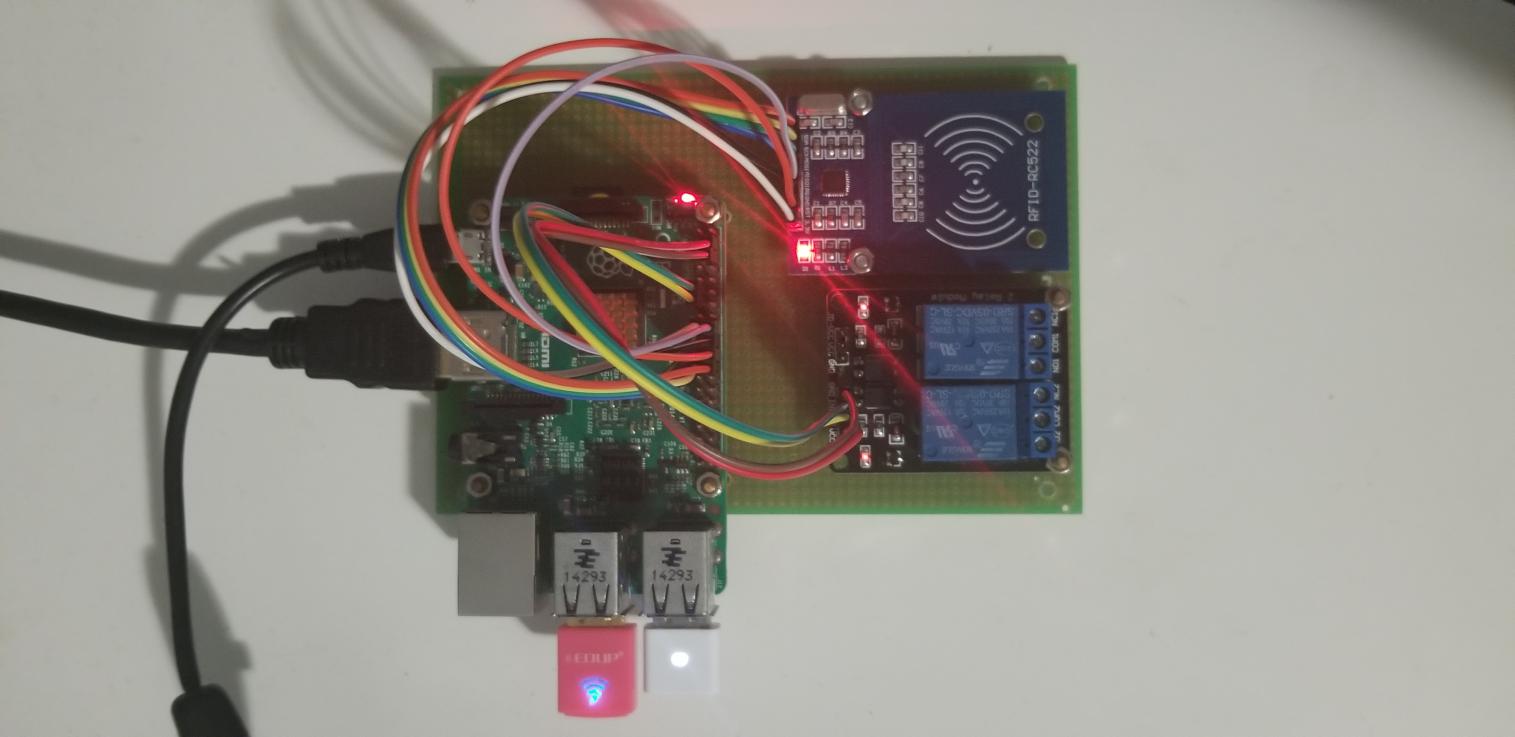
You will see a cisco like CLI UI.



server>acl

Acl>? # check available commands

Entrance raspberry pi setup



This host contains a RFID reader and a Switch(lock) module to simulate the entrance of a company.



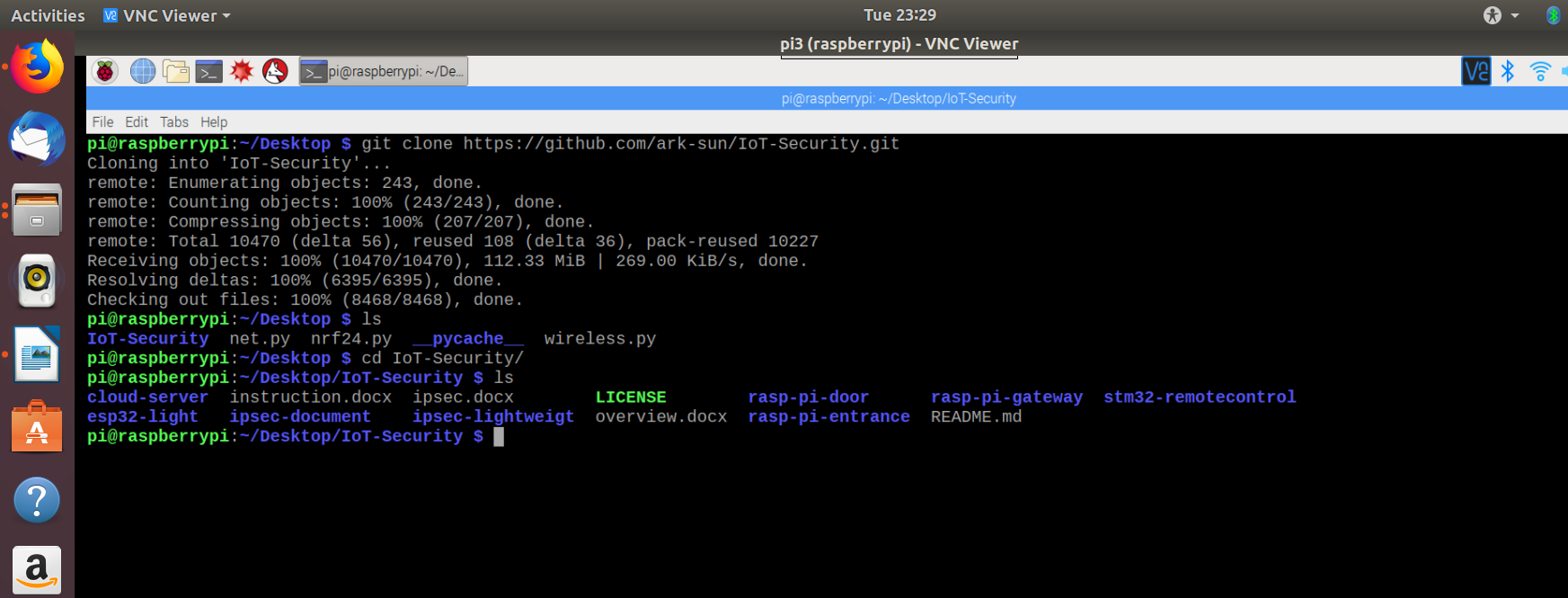
RFID cards.

cd ~/Desktop/

git clone <https://github.com/ark-sun/IoT-Security.git>

Cd IoT-Security/

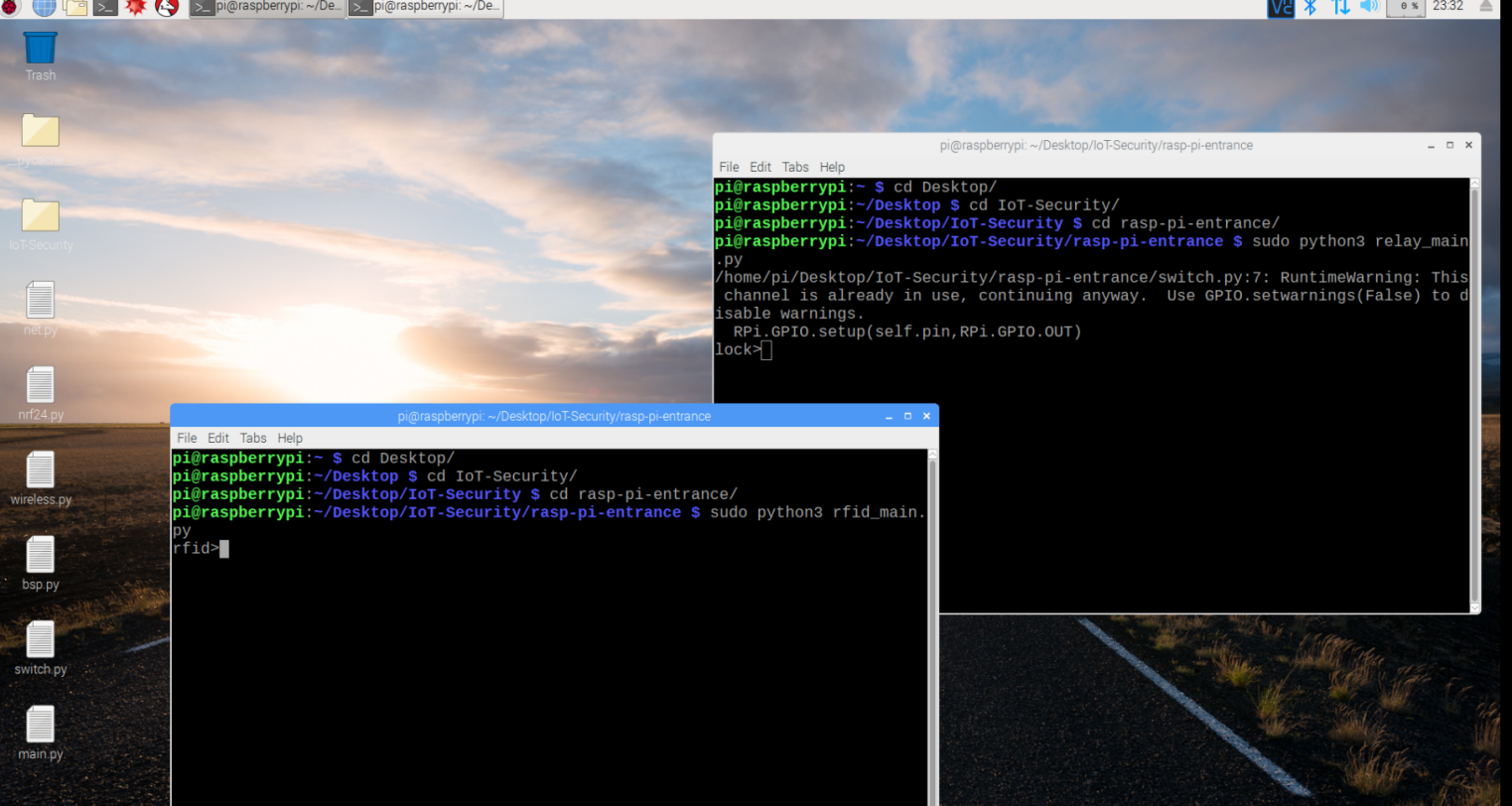
Cd rasp-pi-entrance



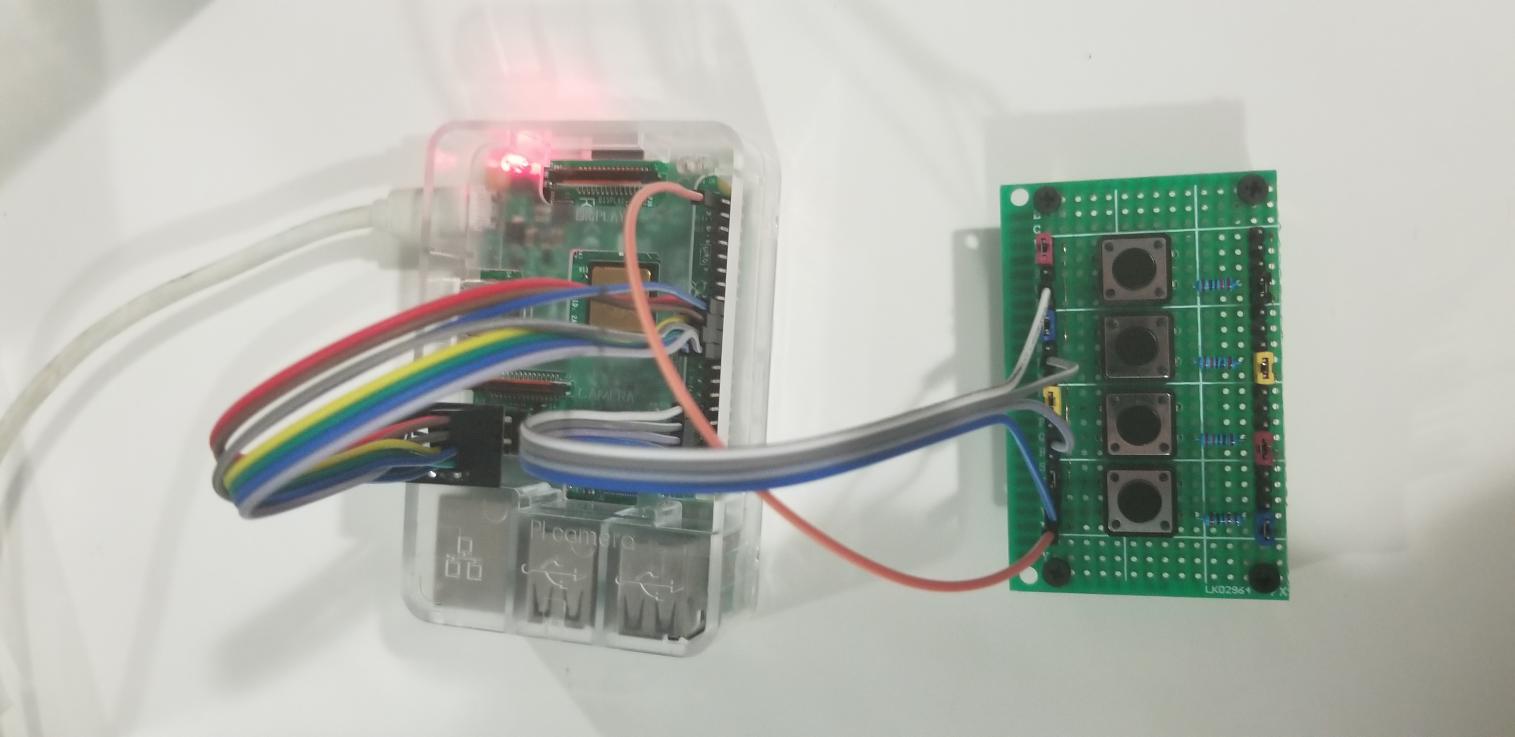
Open two terminals.

Sudo python3 rfid\_main.py

Sudo python3 relay\_main.py



Gateway raspberry pi setup



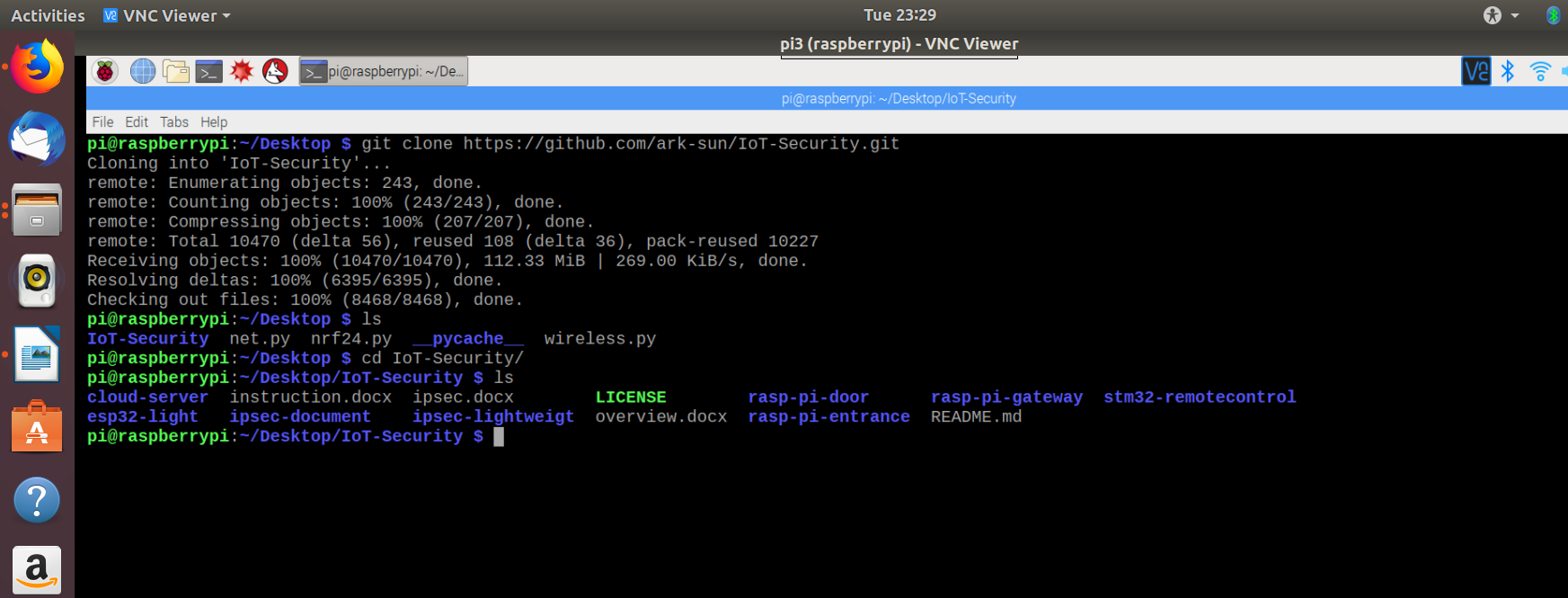
Gateway host contains a 4 buttons keyboard.

cd ~/Desktop/

git clone <https://github.com/ark-sun/IoT-Security.git>

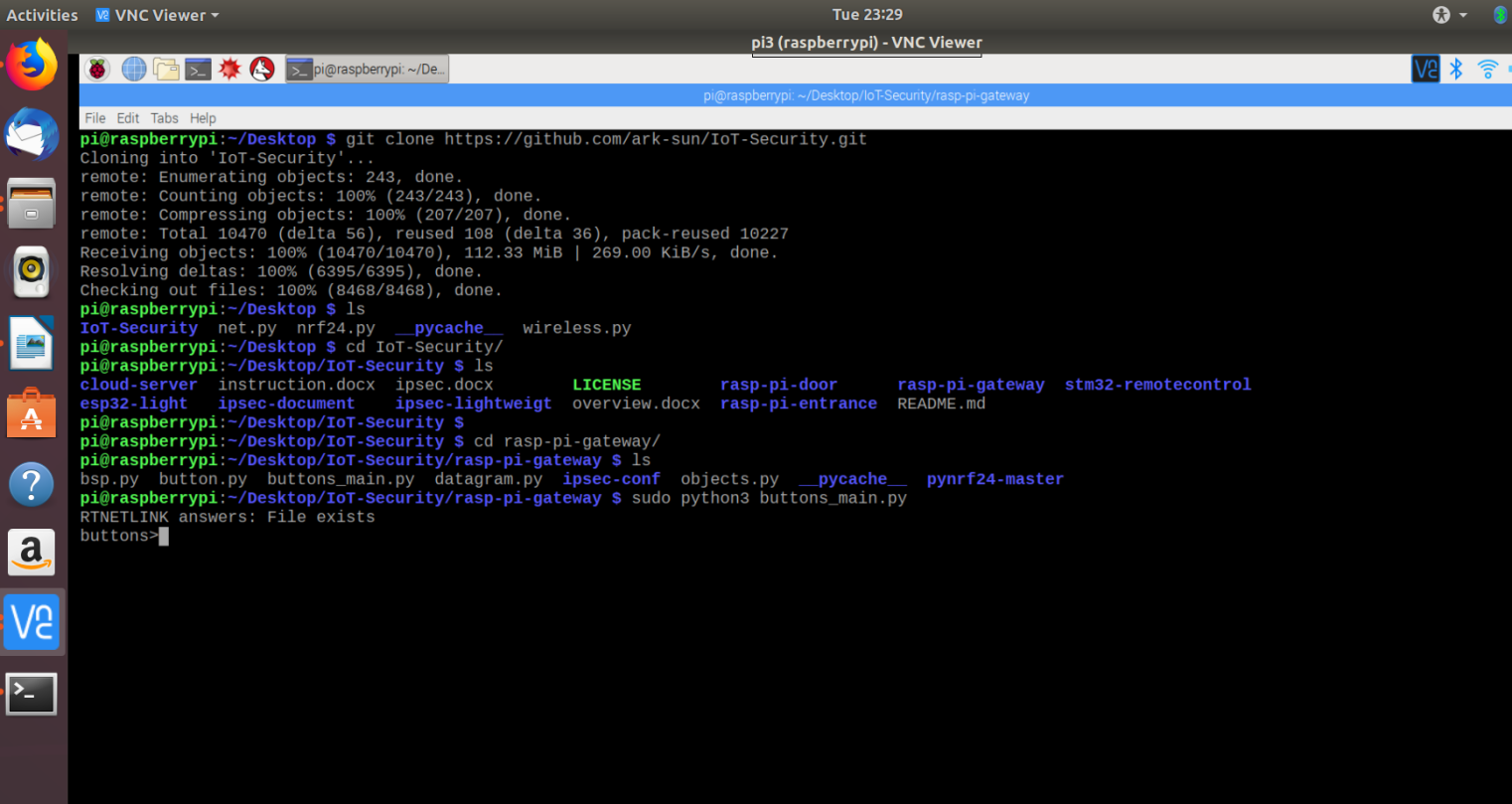
Cd IoT-Security/

Cd rasp-pi-gateway

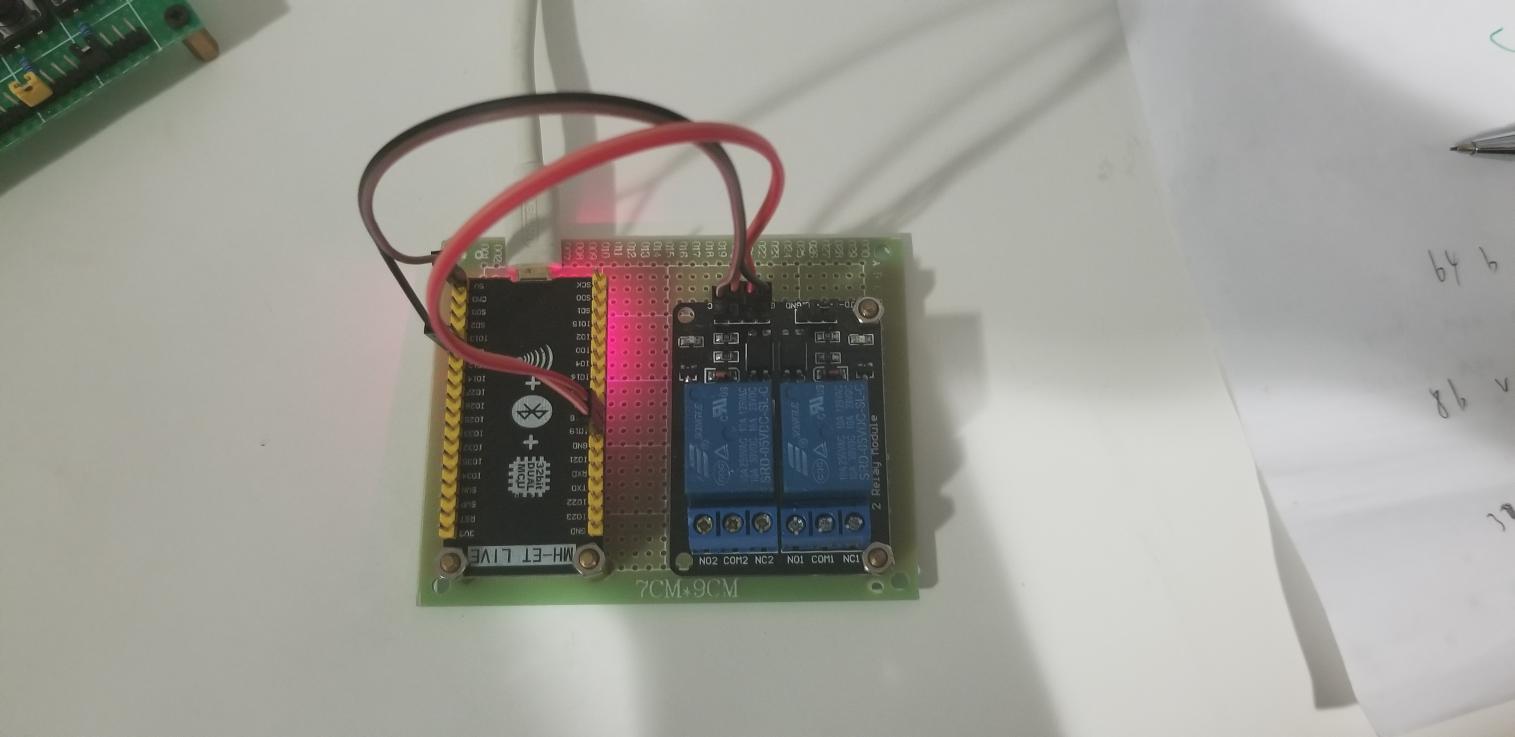


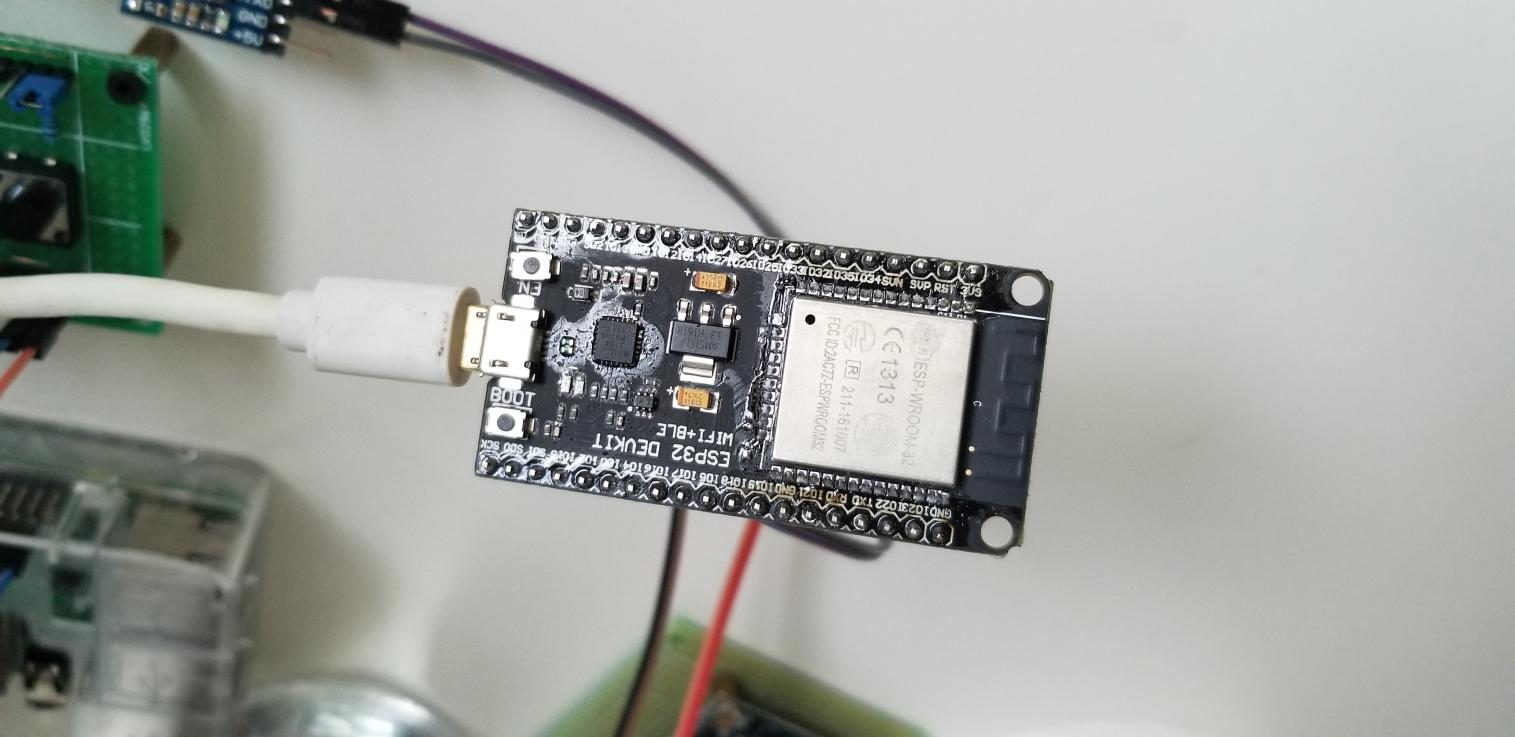
Cd rasp-pi-gateway

Sudo python3 buttons\_main.py



ESP32 tool chain, compile and flash





HOW TO SETUP ESP32 toolchain

Based on Ubuntu 18.04 64bit

Get-started: <https://docs.espressif.com/projects/esp-idf/en/latest/get-started/>

Linux toolchain: <https://docs.espressif.com/projects/esp-idf/en/latest/get-started/linux-setup.html>

(1)utilties

Mkdir ~/Desktop/esp

Sudo apt install git wget make libncurses-dev flex bison gperf python python-serial

(2)xtensa-gcc-compiler:

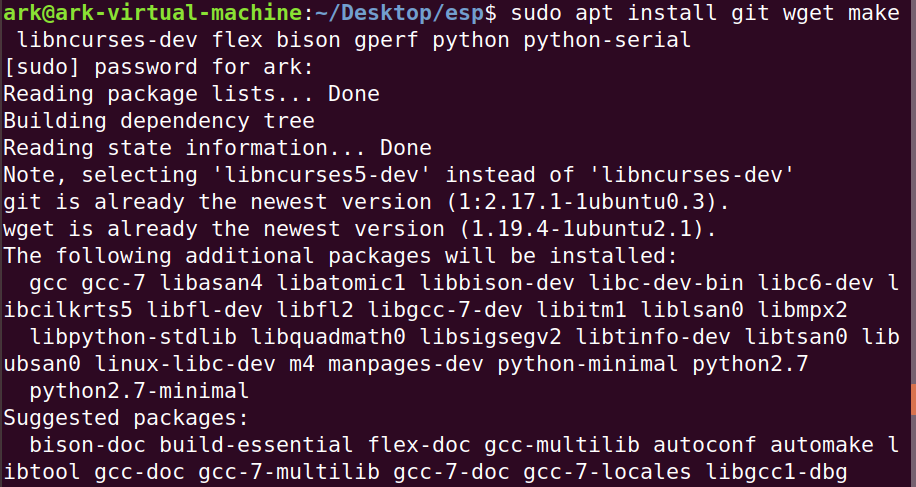
cd ~/Desktop/esp

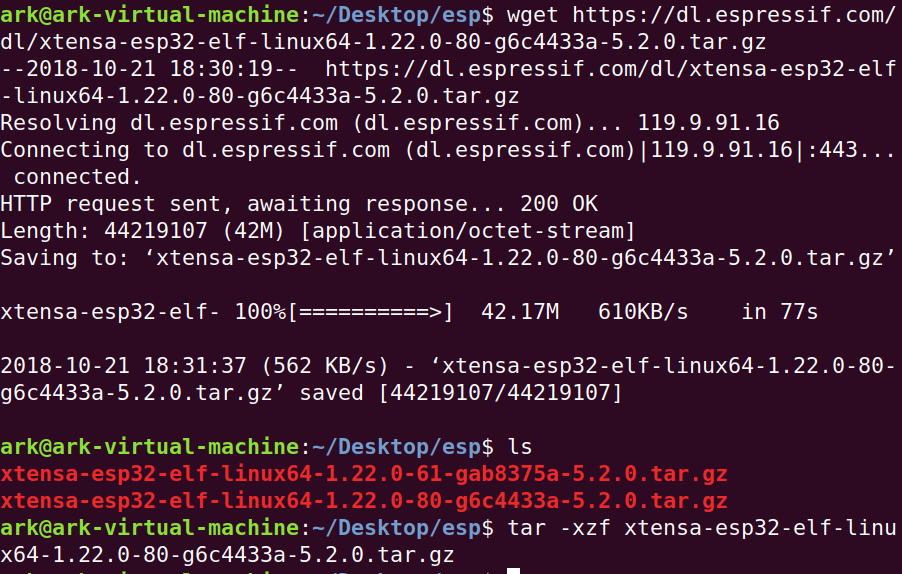
wget <https://dl.espressif.com/dl/xtensa-esp32-elf-linux64-1.22.0-80-g6c4433a-5.2.0.tar.gz>

tar -xzf xtensa-esp32-elf-linux64-1.22.0-80-g6c4433a-5.2.0.tar.gz

# env-var

export PATH="$HOME/Desktop/esp/xtensa-esp32-elf/bin:$PATH"





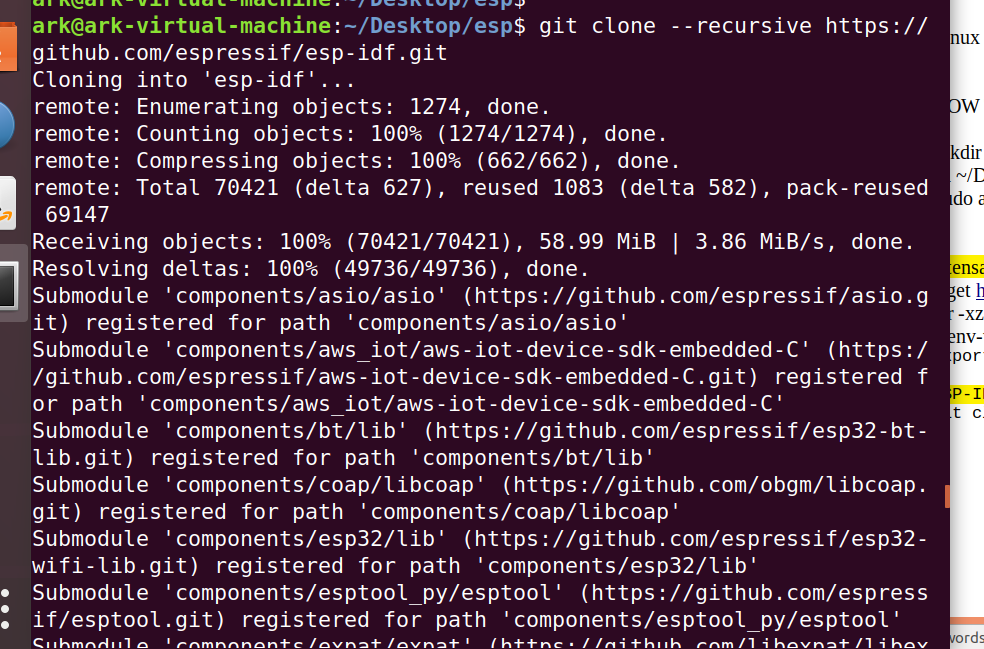
(3)ESP-IDF lib:

cd ~/Desktop/esp

git clone --recursive https://github.com/espressif/esp-idf.git

# env-var

export IDF\_PATH=~/Desktop/esp/esp-idf

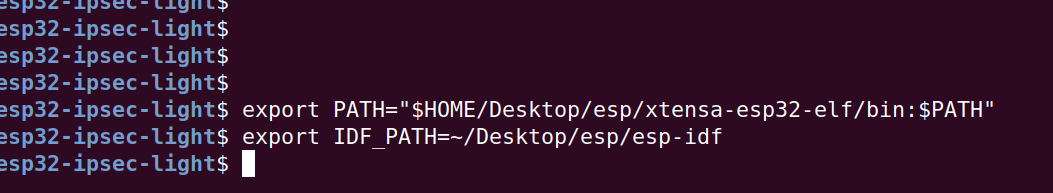


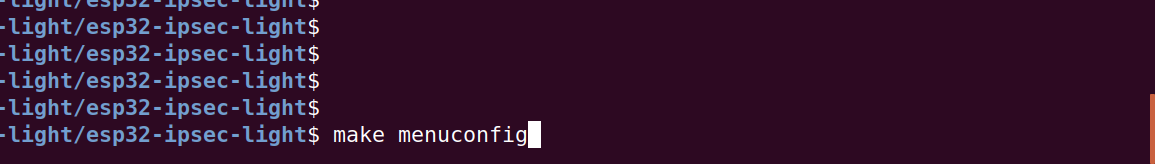
(4)compile project

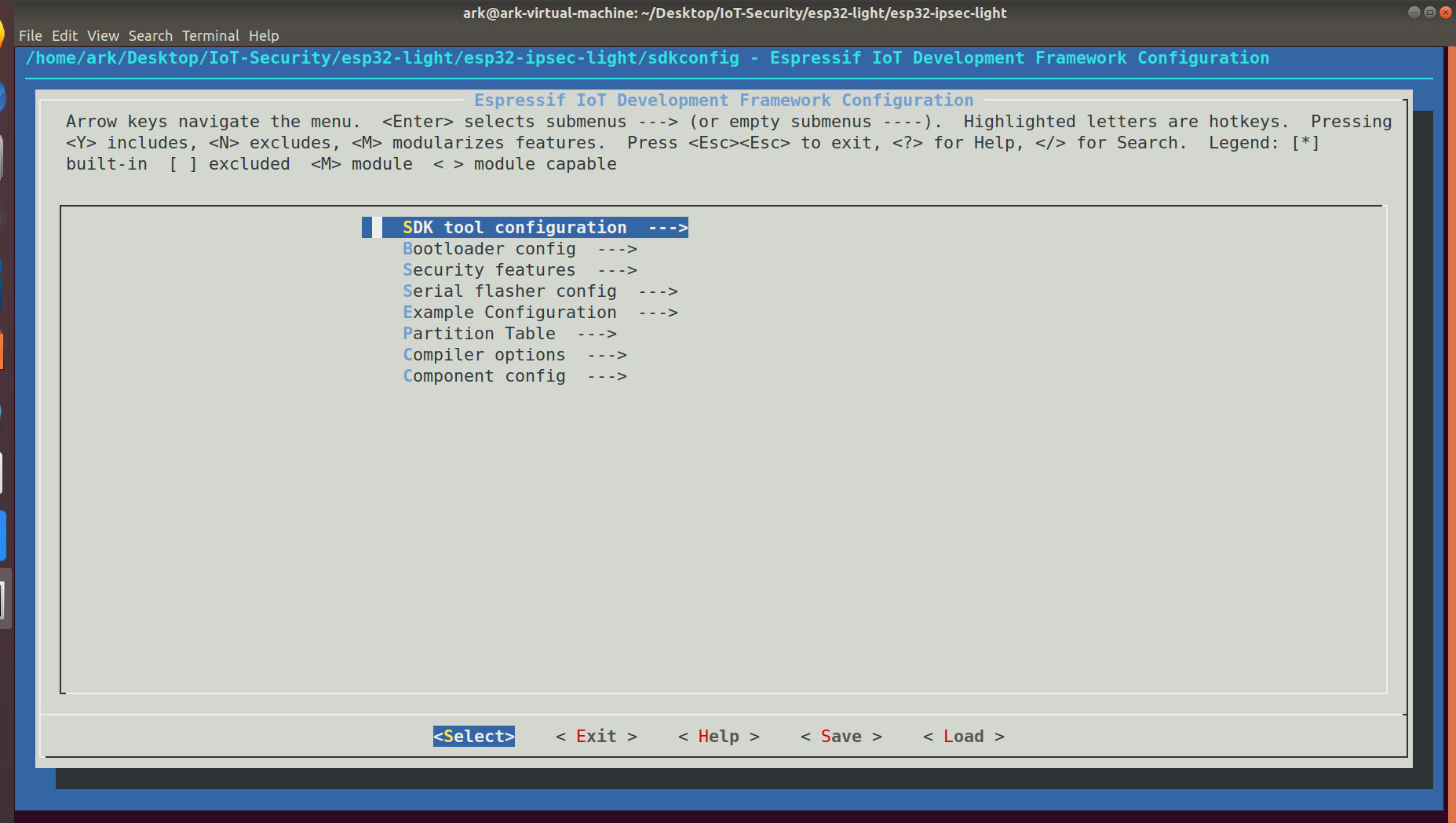
export PATH="$HOME/Desktop/esp/xtensa-esp32-elf/bin:$PATH"

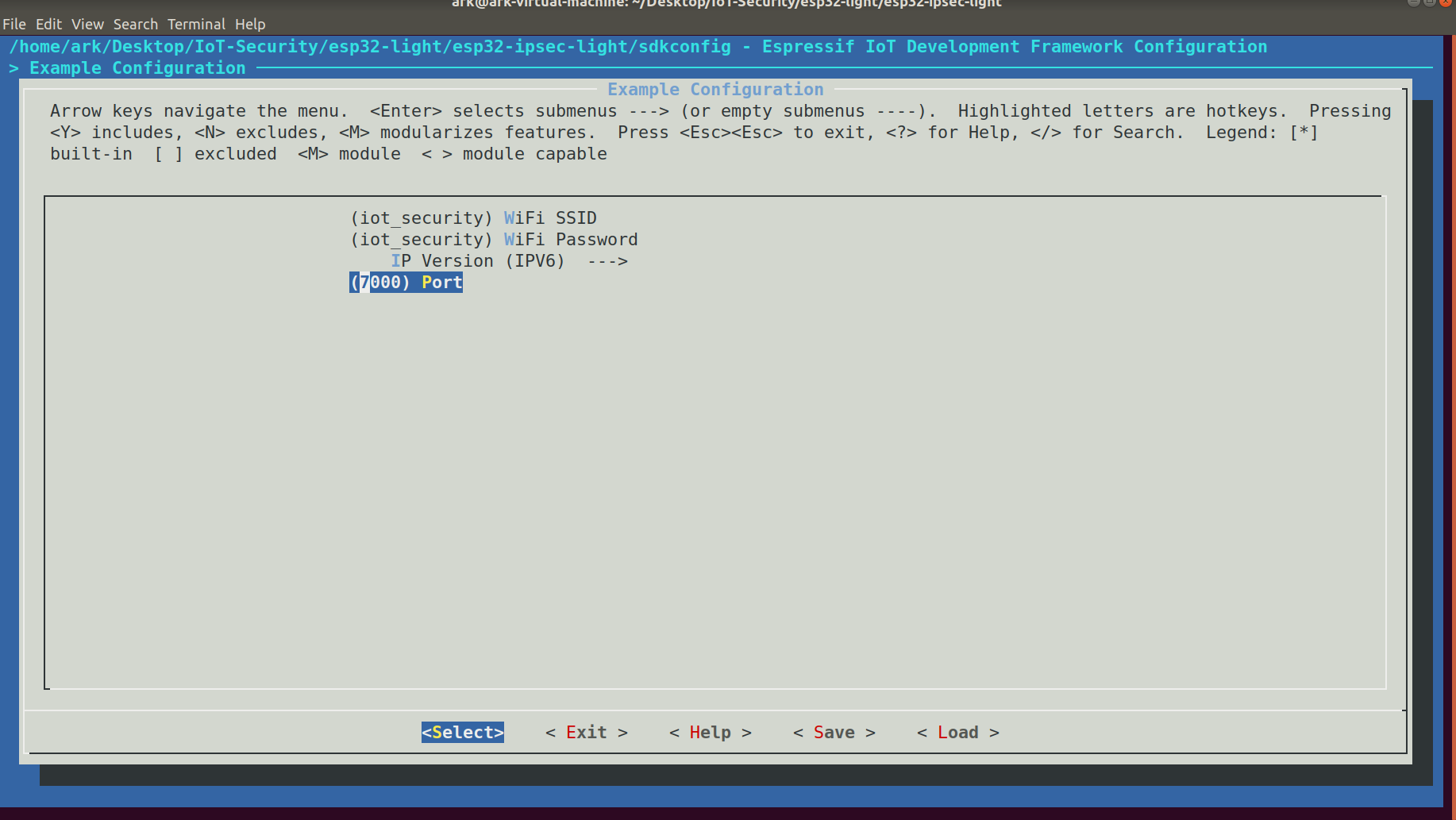
export IDF\_PATH=~/Desktop/esp/esp-idf

make menuconfig







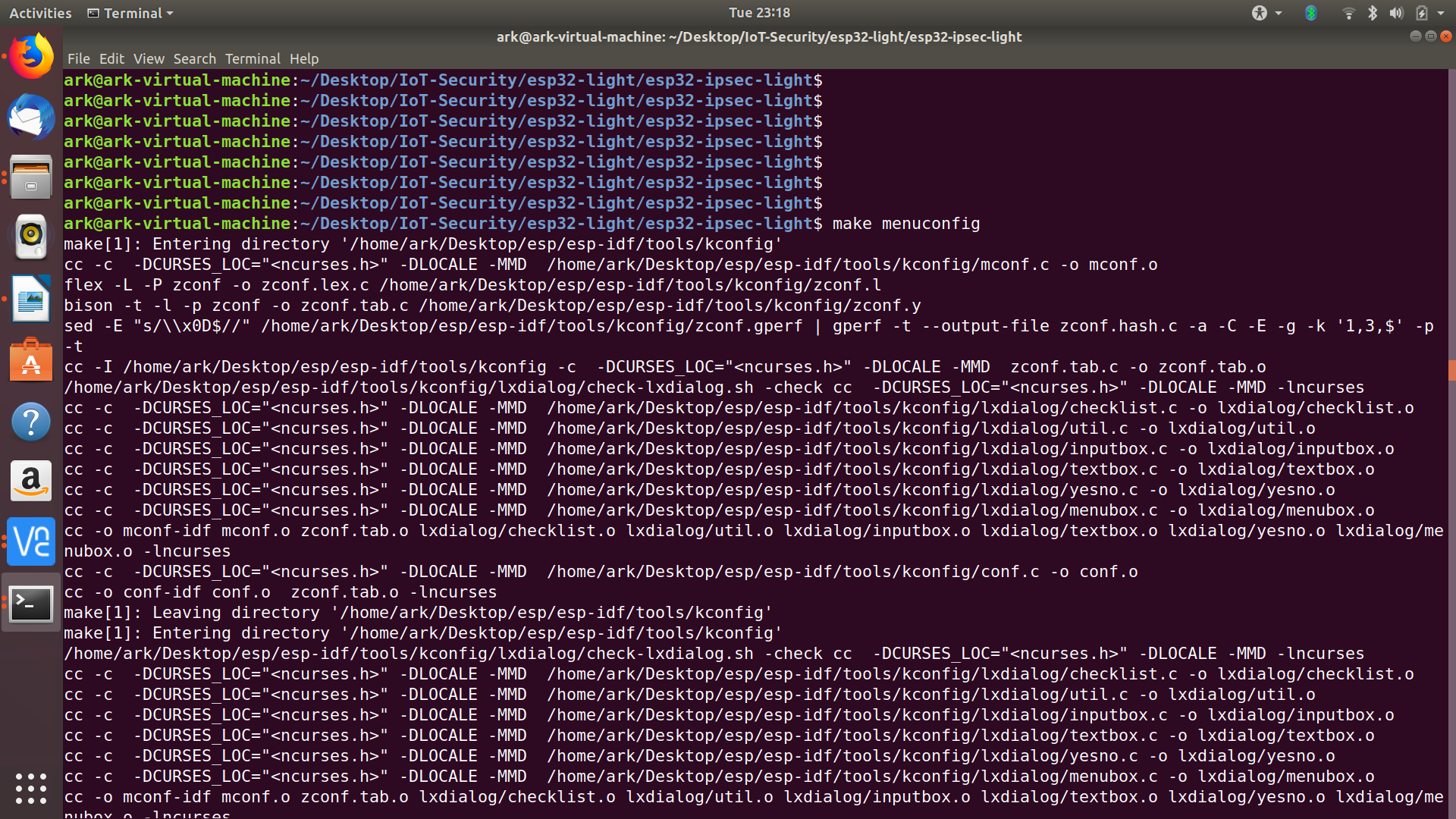


export PATH="$HOME/Desktop/esp/xtensa-esp32-elf/bin:$PATH"

export IDF\_PATH=~/Desktop/esp/esp-idf

make flash # compile and flash

make monitor



Amend LwIP:

**lwip/src/include/lwip/socket.h**

#define IPPROTO\_IP 0

#define IPPROTO\_ICMP 1

#define IPPROTO\_TCP 6

#define IPPROTO\_UDP 17

#define IPPROTO\_ESP 50//add this protocol

#if LWIP\_IPV6

#define IPPROTO\_IPV6 41

#define IPPROTO\_ICMPV6 58

#endif /\* LWIP\_IPV6 \*/

#define IPPROTO\_UDPLITE 136

#define IPPROTO\_RAW 255

**lwip/src/include/lwip/prot/ip6.h**

#define IP6\_NEXTH\_HOPBYHOP 0

#define IP6\_NEXTH\_TCP 6

#define IP6\_NEXTH\_UDP 17

#define IP6\_NEXTH\_ENCAPS 41

#define IP6\_NEXTH\_ROUTING 43

#define IP6\_NEXTH\_FRAGMENT 44

#define IP6\_NEXTH\_ESP 50//add this

#define IP6\_NEXTH\_ICMP6 58

#define IP6\_NEXTH\_NONE 59

#define IP6\_NEXTH\_DESTOPTS 60

#define IP6\_NEXTH\_UDPLITE 136

**lwip/src/core/ipv6/ip6.c**

#if LWIP\_ICMP6

case IP6\_NEXTH\_ICMP6:

/\* Point to payload. \*/

pbuf\_header(p, -(s16\_t)ip\_data.current\_ip\_header\_tot\_len);

icmp6\_input(p, inp);

break;

#endif /\* LWIP\_ICMP \*/

#if 1// add this part

case IP6\_NEXTH\_ESP:

/\* Point to payload. \*/

pbuf\_free(p);

break;

#endif /\* LWIP\_ICMP \*/

default:

#if LWIP\_ICMP6

/\* send ICMP parameter problem unless it was a multicast or ICMPv6 \*/

if ((!ip6\_addr\_ismulticast(ip6\_current\_dest\_addr())) &&

(IP6H\_NEXTH(ip6hdr) != IP6\_NEXTH\_ICMP6)) {

icmp6\_param\_problem(p, ICMP6\_PP\_HEADER, ip\_data.current\_ip\_header\_tot\_len - hlen);

