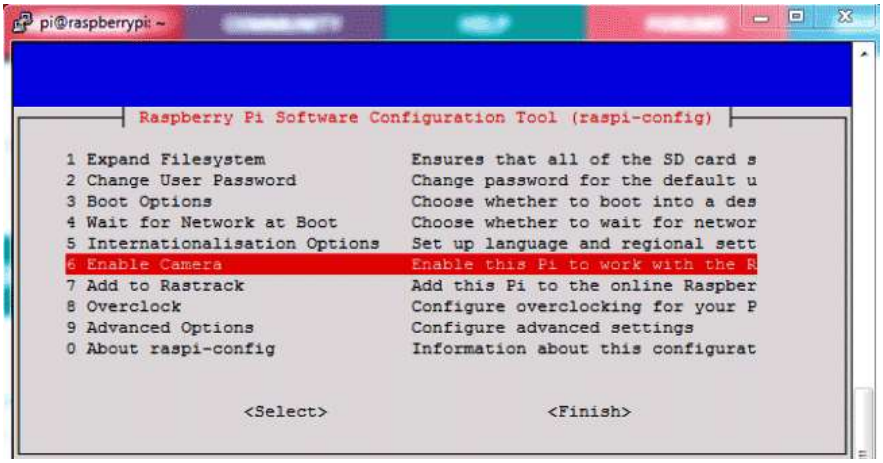
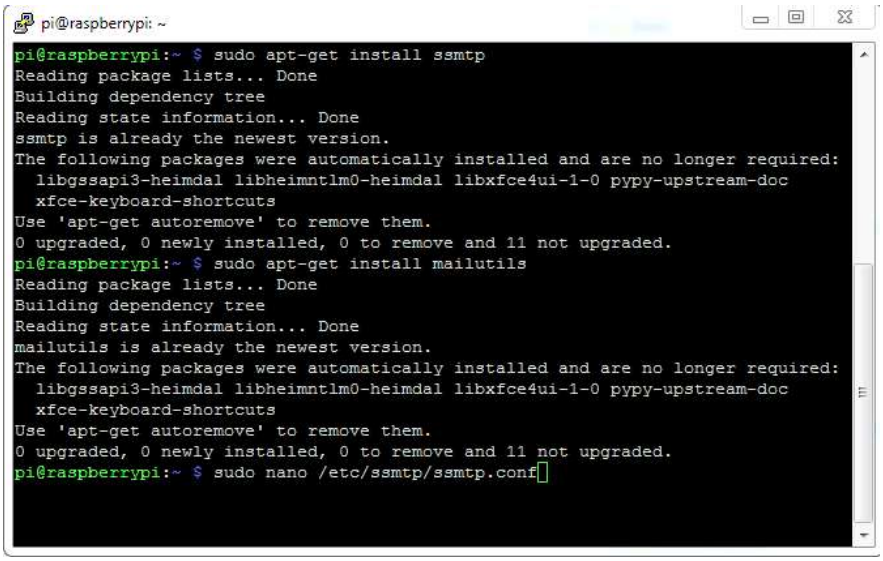


Then user needs to **reboot Raspberry Pi**, by issuing *sudo reboot*, so that new setting can take. Now your Pi camera is ready to use.

Now after setting up the Pi Camera, we will install software for sending the mail. Here we are using *ssmtp* which is an easy and good solution for **sending mail using command line or using Python Script**. We need to install two Libraries for sending mails using SMTP:

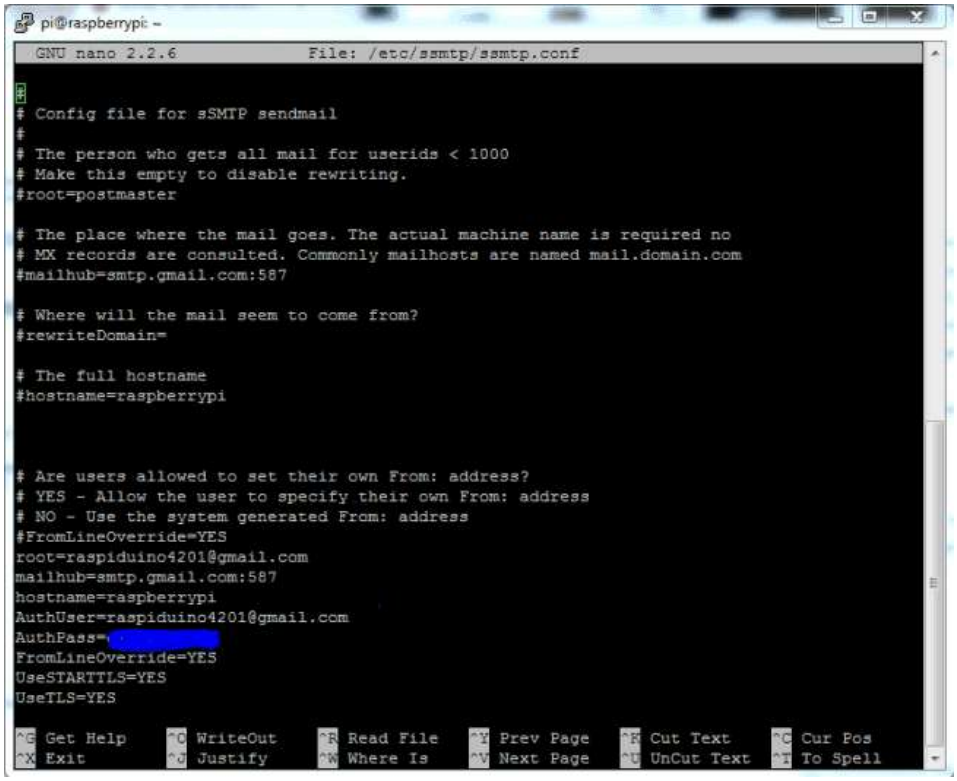
```
sudo apt-get install ssmtp
sudo apt-get install mailutils
```



After installing libraries, user needs to open *ssmtp.conf* file and edit this configuration file as shown in the Picture below and then save the file. To save and exit the file, Press 'CTRL+x', then 'y' and then press 'enter'.

```
sudo nano /etc/ssmtp/ssmtp.conf
```

```
root=YourEmailAddress
mailhub=smtp.gmail.com:587
hostname=raspberrypi
AuthUser=YourEmailAddress
AuthPass=YourEmailPassword
FromLineOverride=YES
UseSTARTTLS=YES
UseTLS=YES
```



We can also **test it by sending a test mail** by issuing below command, you shall get the mail on the mentioned email address if everything is working fine:

```
echo "Hello saddam" | mail -s "Testing..." saddam4201@gmail.com
```

The Python Program of this project plays a very important role to perform all the operations. First of all, we include required libraries for email, initialize variables and define pins for PIR, LED and other components. For sending simple email, smtplib (https://docs.python.org/2/library/smtplib.html) is enough but if you want to send mail in cleaner way with subject line, attachment etc. then you need to use MIME (Multipurpose Internet Mail Extensions).

```
import RPi.GPIO as gpio
import picamera
import time
import smtplib
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders
from email.mime.image import MIMEImage
```

After it, we have initialized mail and define mail address and messages:

```
fromaddr = "raspiduino4201@gmail.com"
toaddr = "saddam4201@gmail.com"
mail = MIMEMultipart()
mail['From'] = fromaddr
mail['To'] = toaddr
mail['Subject'] = "Attachment"
body = "Please find the attachment"
```

Then we have created *def sendMail(data)* function for sending mail:

```
def sendMail(data):
    mail.attach(MIMEText(body, 'plain'))
    print data
    dat='%s.jpg'%data
    print dat
    attachment = open(dat, 'rb')
    image=MIMEImage(attachment.read())
    attachment.close()
    mail.attach(image)
    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.starttls()
    server.login(fromaddr, "your password")
    text = mail.as_string()
    server.sendmail(fromaddr, toaddr, text)
    server.quit()
```

Function *def capture_image()* is created to **capture the image of intruder** with time and date.

```
def capture_image():
    data= time.strftime("%d_%b_%Y|%H:%M:%S")
    camera.start_preview()
    time.sleep(5)
    print data
    camera.capture('%s.jpg'%data)
    camera.stop_preview()
    time.sleep(1)
    sendMail(data)
```

Then we initialized the **Picamera** with some of its settings:

```
camera = picamera.PiCamera()
camera.rotation=180
camera.awb_mode= 'auto'
camera.brightness=55
```

And now in last, we have read PIR sensor output and when its goes high Raspberry Pi calls the *capture_image()* function to capture the image of intruder and send a alert message with the picture of intruder as an attachment. We have used *sendmail()* insdie *capture_image()* function for sending the mail.

```
while 1:
    if gpio.input(pir)==1:
        gpio.output(led, HIGH)
        capture_image()
        while(gpio.input(pir)==1):
            time.sleep(1)

    else:
        gpio.output(led, LOW)
        time.sleep(0.01)
```

So this how this **Raspberry Pi Security System** works, you can also use Ultrasonic sensor (http://circuitdigest.com/microcontroller-projects/door-alarm-using-arduino-ultrasonic-sensor) or IR sensor (http://circuitdigest.com/electronic-circuits/ir-security-alarm-circuit) to detect the presence of burglar or intruder. Further check the **Full Code** and demonstration **Video** below.

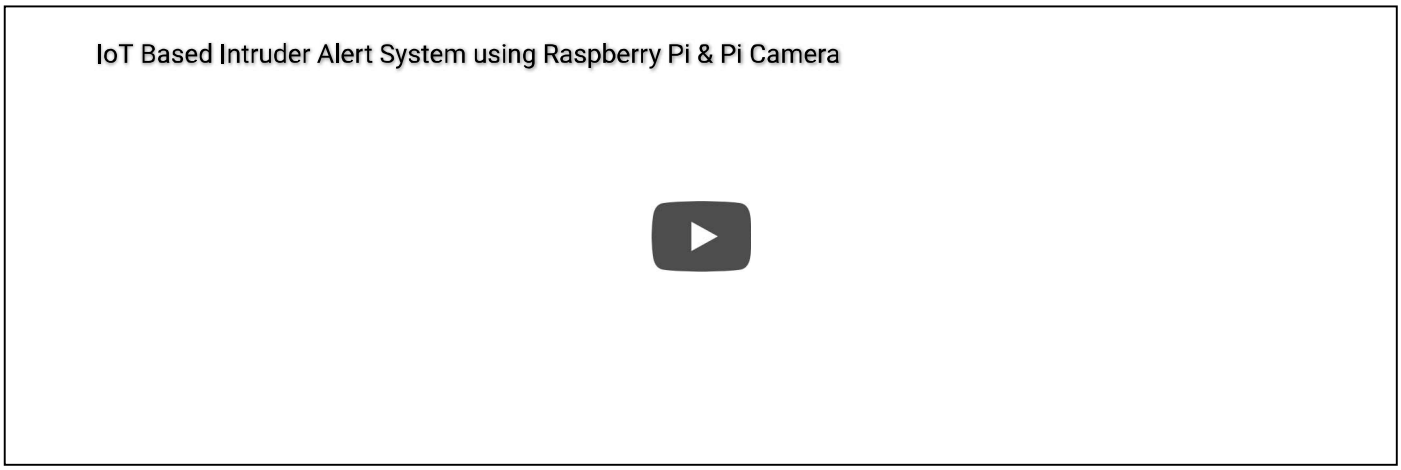
Code

```
import RPi.GPIO as gpio
import picamera
import time
import smtplib
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders
```

```
from email.mime.image import MIMEImage

fromaddr = "raspiduino4201@gmail.com (mailto:raspiduino4201@gmail.com)"    # change the email address accordingly
toaddr = "saddam4201@gmail.com (mailto:saddam4201@gmail.com)"

mail = MIMEMultipart()

mail['From'] = fromaddr
mail['To'] = toaddr
mail['Subject'] = "Attachment"
body = "Please find the attachment"
led=17
pir=18
HIGH=1
LOW=0
gpio.setwarnings(False)
gpio.setmode(gpio.BCM)
gpio.setup(led, gpio.OUT)          # initialize GPIO Pin as outputs
gpio.setup(pir, gpio.IN)          # initialize GPIO Pin as input
data=""
def sendMail(data):
  mail.attach(MIMEText(body, 'plain'))
  print data
  dat='%s.jpg'%data
  print dat
  attachment = open(dat, 'rb')
  image=MIMEImage(attachment.read())
  attachment.close()
  mail.attach(image)
  server = smtplib.SMTP('smtp.gmail.com', 587)
  server.starttls()
  server.login(fromaddr, "your password")
  text = mail.as_string()
  server.sendmail(fromaddr, toaddr, text)
  server.quit()
def capture_image():
  data= time.strftime("%d_%b_%Y|%H:%M:%S")
  camera.start_preview()
  time.sleep(5)
  print data
  camera.capture('%s.jpg'%data)
  camera.stop_preview()
  time.sleep(1)
  sendMail(data)
gpio.output(led , 0)
camera = picamera.PiCamera()
camera.rotation=180
camera.awb_mode= 'auto'
camera.brightness=55
while 1:
  if gpio.input(pir)==1:
    gpio.output(led, HIGH)
    capture_image()
    while(gpio.input(pir)==1):
```

```
        time.sleep(1)

else:
    gpio.output(led, LOW)
    time.sleep(0.01)
```

Video



TAGS RASPBERRY PI (/TAGS/RASPBERRY-PI) PIR SENSOR (/TAGS/PIR-SENSOR) PI CAMERA (/TAGS/PI-CAMERA) HOME SECURITY (/TAGS/HOME-SECURITY) IOT (/TAGS/IOT) SENSORS (/TAGS/SENSORS)