```
#include <Servo.h>
#include <LiquidCrystal.h>
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 4; //three columns
char keys[ROWS][COLS] = {
  {'7','8','9', '/'},
  {'4','5','6', '*'},
  {'1','2','3', '-'},
  {'o','0','=', '+'}
};
byte rowPins[ROWS] = {31, 33, 35, 37}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {23, 25, 27, 29}; //connect to the column pinouts of the keypad

const int rs = 13, en = 12, d4 = 8, d5 = 9, d6 = 10, d7 = 11;

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
Servo myservo;  // create servo object to control a servo

const byte doorLedPin = 50, buzzPin = 53, servoPin = 7 ;

byte state = 0;
String prompts[] = {"Pass: ", "Door Open", "ChPass: ", "ChTime: ","anti theft"};
String inputBuffer, password = "1234";
bool timerOn = false;
int timerTime = 10;
int anti=5;
```

```
long t0 = millis();
byte wrong=0;
bool theft=false;
void setup() {
  Serial.begin(9600);

  myservo.attach(servoPin);
  myservo.write(0);

  lcd.begin(16, 2);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(prompts[state]);

  pinMode(doorLedPin, OUTPUT);
  pinMode(buzzPin, OUTPUT);

}

void pPrompt() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print(prompts[state]);
  inputBuffer = "";
}

void buzz() {
  digitalWrite(buzzPin, HIGH);
  delay(200);
  digitalWrite(buzzPin, LOW);
}

void open_door() {
  t0 = millis();
  timerOn = true;
  digitalWrite(doorLedPin, HIGH);
  buzz();
  myservo.write(180);
}

void lock_door() {
  myservo.write(0);
  digitalWrite(doorLedPin, LOW);
  timerOn = false;
  state=0;
  pPrompt();
  buzz();
}

float calc_time() {
  if (!timerOn) {
    return 0;
  }
  long nowT = millis();
  float diff = (float) timerTime - ((nowT - t0) / 1000.0);
  if (diff <= 0) {
    lock_door();
    diff = 0;
  }
  return diff;
}
float anti_theft() {
  long nowT = millis();
  float diff;
   if (!theft) {
    return calc_time();
   }   diff = (float) anti- ((nowT - t0) / 1000.0);
  if (diff <= 0) {
    diff = 0;
    anti=anti*2;
    theft=false;
    state=0;
    pPrompt();

  }
  return diff;

}

boolean password_entered() {

  lcd.setCursor(0, 0);
  if (inputBuffer == password) {
    lcd.print("Correct password");
    state = 1;
    open_door();
  } else {
     if(wrong==3){
      wrong=0;
      state=4;
      buzz();
      return true;
     }
    state = 0;
    lcd.print("Wrong password");
    wrong++;
  }
  delay(100);
  return false;
}

void loop() {
  float timer = calc_time();
  timer=anti_theft();

  lcd.setCursor(0, 1);
  lcd.print("Timer: " + String(timer) + "  ");

  char key = keypad.getKey();
  if (key && theft==false){
    lcd.setCursor(prompts[state].length() + inputBuffer.length(), 0);
    if(key!= '=' && key !='-'&&key!='*'&&key!='o')
    lcd.print(key);

    if (key == '=')
    {
      if (!timerOn) {
        // Safe is locked - should enter password
        theft=password_entered();
      } else if (state == 2) {
        // Change password
        password = inputBuffer;
        state = 1;
      } else if (state == 3) {
        // Change timer
        timerTime = inputBuffer.toInt();
        state = 1;
      }
      pPrompt();
    } else if (key == '*' && timerOn) {
      state = 2;
      pPrompt();
    } else if (key == '-' && timerOn) {
      state = 3;
      pPrompt();
    } else if (key == 'o' && timerOn) {
      lock_door();
    } else
      inputBuffer += key;
  }

    delay(50);
}
```