实 验 报 告

课程名称 信息安全

学　　院 软件学院

专　　业 软件工程

姓　　名 于康

学　　号 20302010040

开 课 时 间 2022 至 2023 学年第 二 学期

实验项目名　　称	**DES&AES**	成绩	

一、实验目的

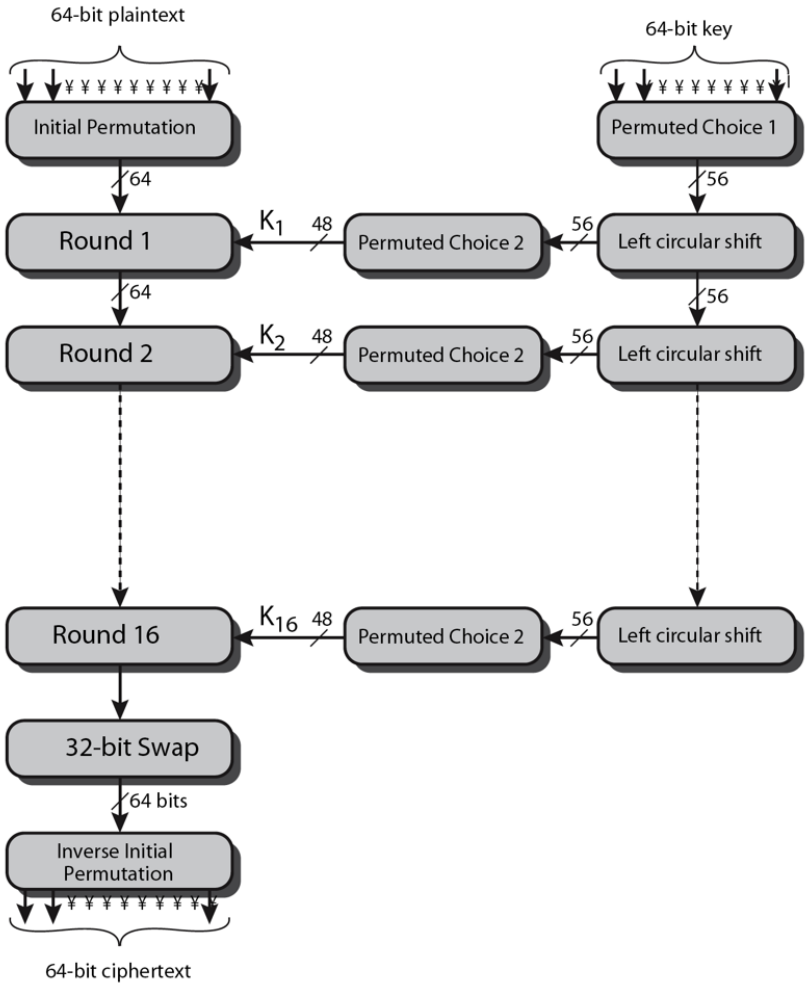
1.实现 DES 算法

2.理解 AES 算法的用途

二、实验内容

1.DES 算法步骤

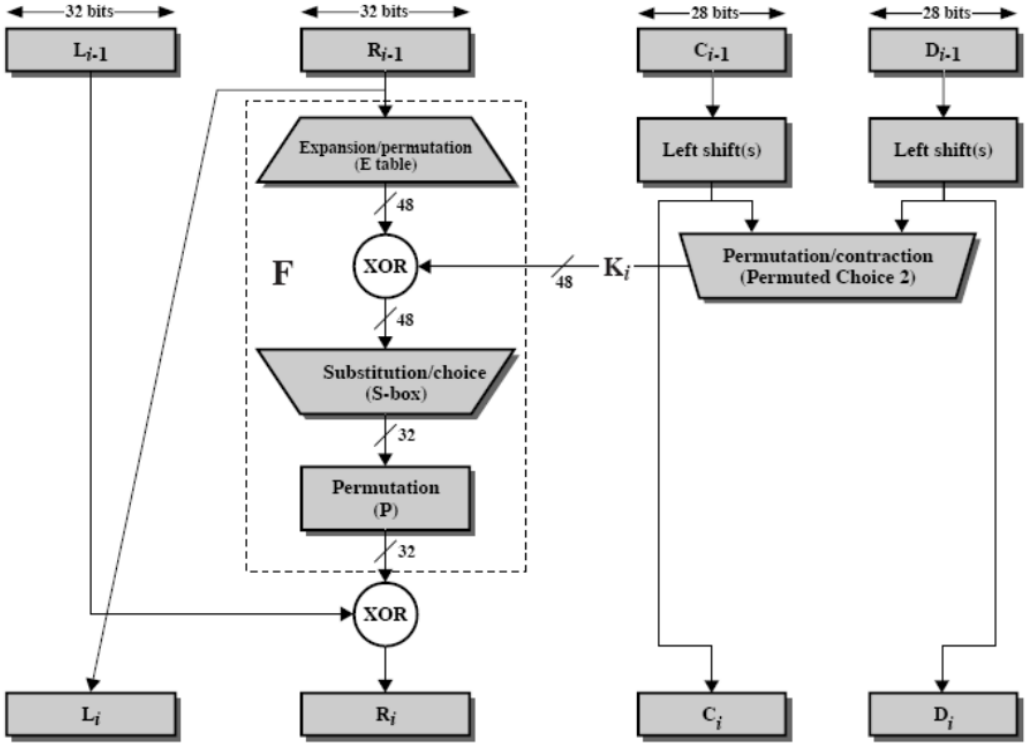
下图展示了 DES 算法的过程，可知经过 PC1、左移、PC2 能够生成 16 组 key，解密是加密的逆，将 key 的应用顺序颠倒。



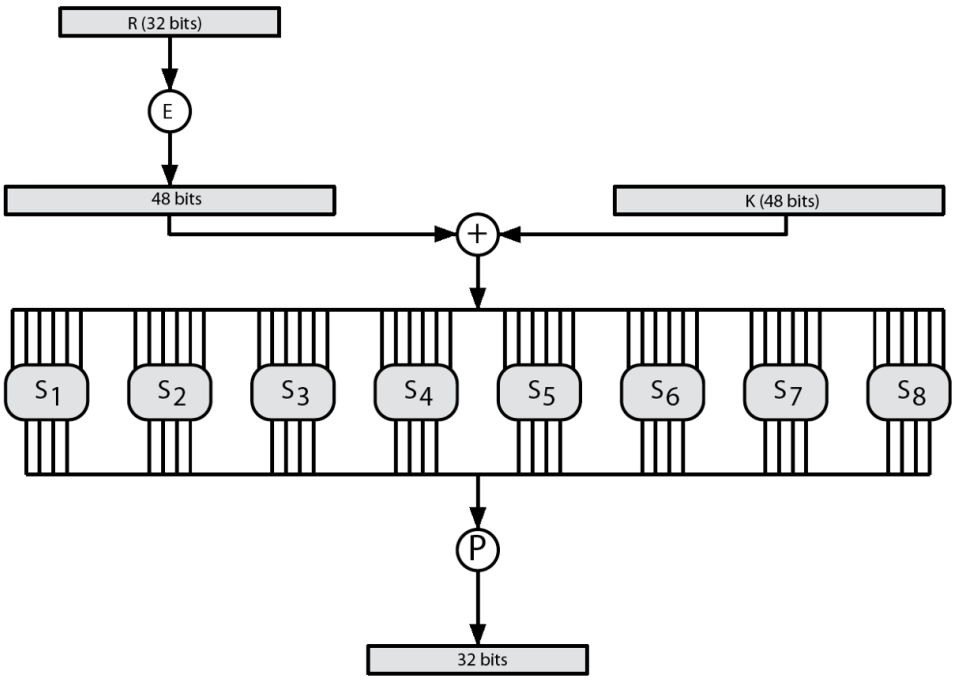
而每一轮对于 64bit 数据块的处理如下图所示，得到公式：

$$L_n = R_{n-1}$$

$$R_n = L_{n-1} + f(R_{n-1}, K_n)$$



其中，f 函数的具体工作过程如下，将 R 扩展后与 K 进行异或操作，之后经过 Sbox 输出为 32bit 数据块，再进行一次置换操作。



2.使用 AES

可以利用 OpenSSL 的 Enc 命令进行加密和解密；可以利用 Python 包 PyCryptodome 提供的 AES 模块。

三、实验步骤

1. permutation_by_table

该函数利用置换表 table 和原数据块及其大小，返回置换后的数据块；而置换表则展示了新块的组成，可见下方代码注释部分：

```
    permuted_block = 0
    for i, bit_index in enumerate(table):
        bit_index -= 1
        # 将原块的第 bit_index 位设置为 permuted_block 的第 i 位
        if block & (1 << (block_len - bit_index - 1)):
            permuted_block |= (1 << (len(table) - i - 1))
    return permuted_block
```

2. generate_round_keys

该函数的作用是生成 16 轮迭代中所需要的 16 个 K。函数输入已经是经过 PC1 后的 28bit 的 C0 和 D0，首先按左移表进行左移：

```
    for i, r_bits in enumerate(lrot_values):
        C0 = lrot(C0, r_bits, 28)
        D0 = lrot(D0, r_bits, 28)
        round_keys[i + 1] = (C0, D0)
```

之后进行 PC2 得到最终使用的 K：

```
    for i, (Ci, Di) in round_keys.items():
        key = (Ci << 28) | Di
        round_keys[i] = permutation_by_table(key, 56, PC2)
```

3. round_function

该函数实现了 DES 算法中的 f 函数，函数输入为 Ri 和 Ki，输出 f 函数结果。

首先是对 Ri 的扩展（E）操作：

```
    Ri = permutation_by_table(Ri, 32, E)
```

之后进行异或运算：

```
    Ri = Ri ^ Ki
```

进入 Sbox 前要将数据块分为 8 组，每组 6 位：

```
    blocks = [Ri >> (6 * i) & 0b111111 for i in range(7, -1, -1)]
```

经过 Sbox 的规则是“第一位和最后一位选行数，中间四位选列数”：

```
    sbox_blocks = []
    for i, block in enumerate(blocks):
        row = (block >> 4 & 0b10) | (block & 0b1)
        col = block >> 1 & 0b1111
```

```
            sbox_blocks.append(Sboxes[i][row * 16 + col])
```

再将新数据组进行拼接：

```
    Ri = 0
    for i in range(8):
        Ri |= sbox_blocks[i] << (4 * (7 - i))
```

最后进行一次置换：

```
    Ri = permutation_by_table(Ri, 32, P)
```

4. encrypt

加密函数补充部分主要为每一轮的过程代码，即下述公式：

$$L_n = R_{n-1}$$

$$R_n = L_{n-1} + f(R_{n-1}, K_n)$$

并且注意到解密是加密的逆，key 的使用顺序不同：

```
    for i in range(1, 17):
        key_index = 0
        if decrypt:
            key_index = 17 - i
        else:
            key_index = i
        temp = L_last
        L_last = R_last
        R_last = temp ^ round_function(R_last, round_keys[key_index])
```

5. 解密 DES 密文

运行程序，得到 cipher_text_even 的解密结果为 *LJingTao*

```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\yuki\Desktop\files\课程\信息安全\lab\lab2> python .\20302010040_des.py
Plain Text: LJingTao
```

6. 解密 AES 密文

用第 5 步 DES 解密的结果作为 AES 的密钥，来解密 AES 密文，得到结果为：

The Advanced Encryption Standard, also known by its original name Rijndael, is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology in 2001. AES is a subset of the Rijndael block cipher developed by two Belgian cryptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

```
root@DESKTOP-M59KV5D:/data/lab2# ls
 20302010040_des.py   even.enc  '实验二(中).pdf'   实验报告模板.docx
 des_scaffold.py      odd.enc    实验二.pdf
root@DESKTOP-M59KV5D:/data/lab2# openssl enc -d -aes256 -in even.enc -out even.plain
enter aes-256-cbc decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
root@DESKTOP-M59KV5D:/data/lab2# cat even.plain
The Advanced Encryption Standard, also known by its original name Rijndael, is a specification
for the encryption of electronic data established by the U.S. National Institute of Standards a
nd Technology in 2001. AES is a subset of the Rijndael block cipher developed by two Belgian cr
yptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal to NIST during the AES s
election process. Rijndael is a family of ciphers with different key and block sizes. For AES,
NIST selected three members of the Rijndael family, each with a block size of 128 bits, but thr
ee different key lengths: 128, 192 and 256 bits.
root@DESKTOP-M59KV5D:/data/lab2#
```

四、实验结果及分析

由上述实验过程可知，解密 DES 密文结果为 *LJingTao*。

利用 DES 解密的结果作为 AES 的密钥，来解密 AES 密文，得到结果为：

The Advanced Encryption Standard, also known by its original name Rijndael, is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology in 2001. AES is a subset of the Rijndael block cipher developed by two Belgian cryptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

（该段落内容为 AES 的相关介绍）

五、实验总结

该实验进行了 DES 算法的实验以及 DES 密文、AES 密文的解密。

对于 DES 算法的整个过程有了更加清晰的理解，包括每一轮的计算过程、16 个 K 的生成、f 函数的实现等，将课堂中的 DES 图示转化成了具体的代码。

通过 OpenSSL 对于 AES 的使用有了一定了解，具体为利用 Enc 进行 AES 的加密和解密操作。