**实验一 AES密码算法**

**姓名：\_\_\_\_\_\_\_\_\_梁鑫嵘\_\_\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_\_200110619\_\_\_\_\_\_\_\_**

1. **运行截图**

*分别截取3组测试结果, 每组截图内容包括明文，密钥，和对应密钥加密的密文和10轮密钥的结果，以及对应解密后的明文。*

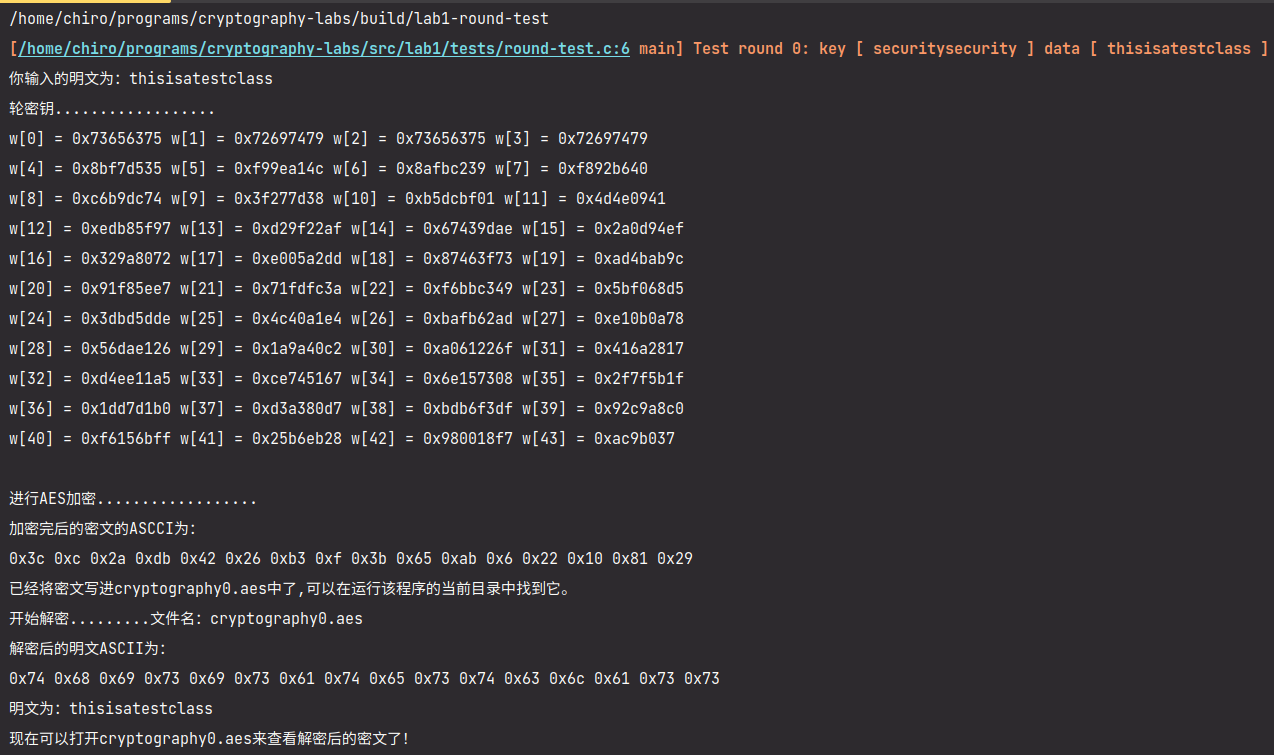
*其中一组明文为thisisatestclass,密钥为securitysecurity*

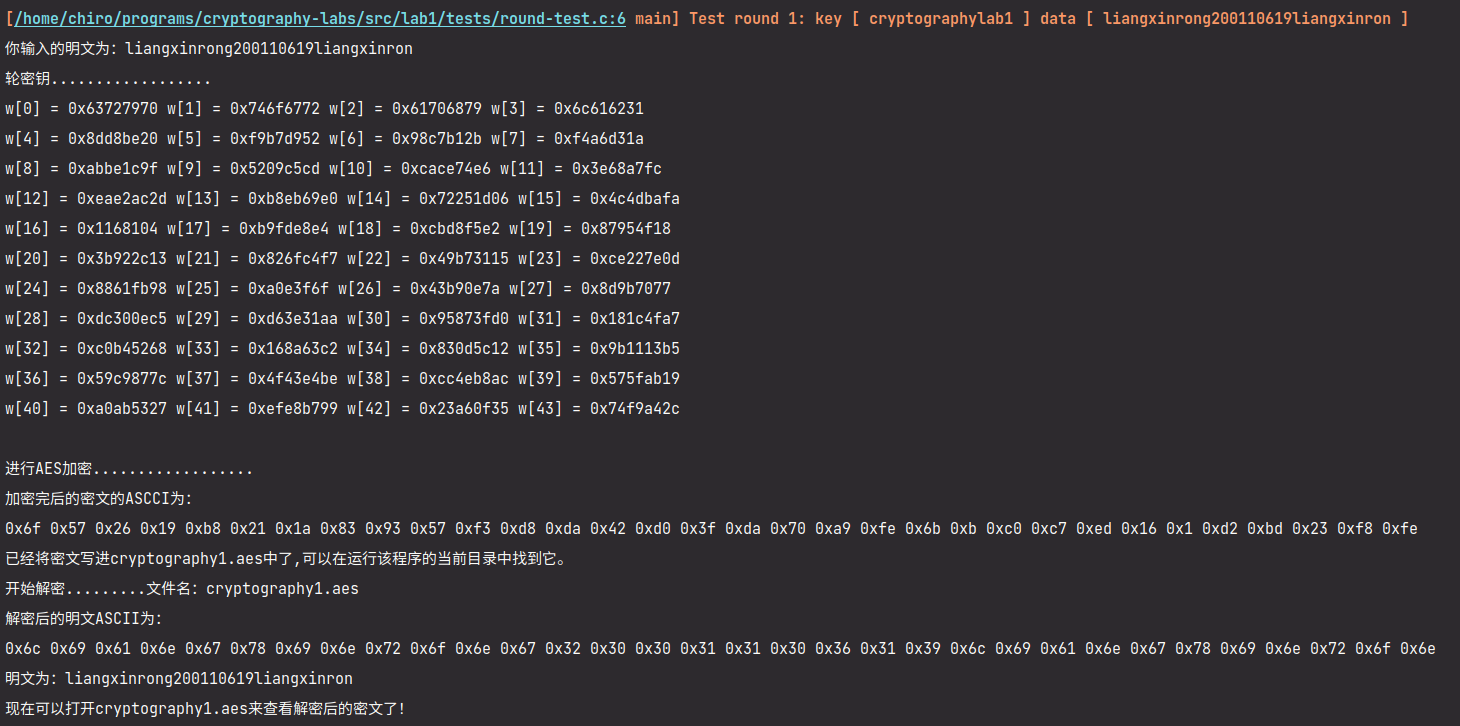
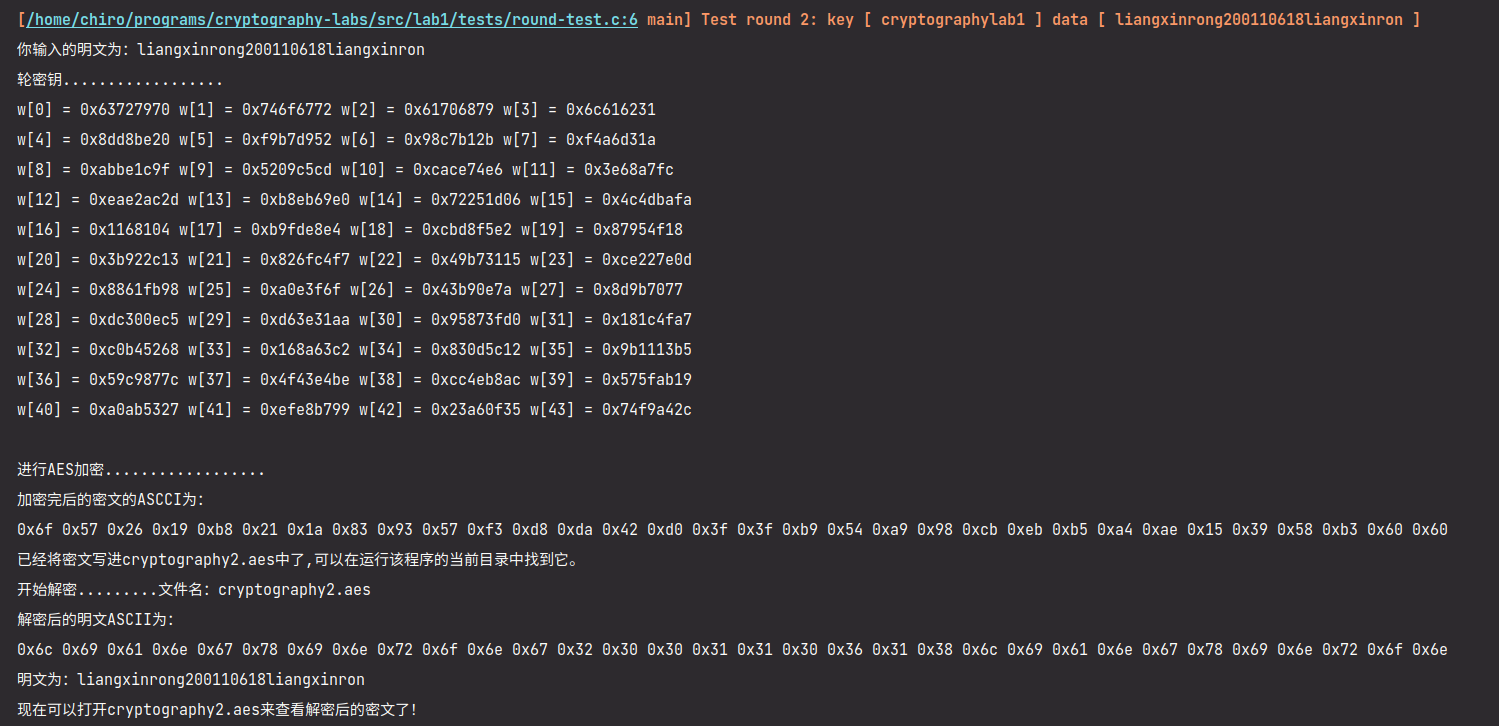
*其他两组明文不同，密钥相同：*

*明文1： 姓名拼音+学号，不足16个字符，重复补齐，例如：suting20188197su*

*明文2：姓名拼音+（学号-1），不足16个字符，重复补齐，例如：suting20188196su*

*密钥为：cryptographylab1*

**使用 EBC 模式：**



**使用 CBC 模式：**

**➜ chiro@chiro-pc ~/programs/cryptography-labs/build git:(master) ✗** ./lab1-round-test

**[/home/chiro/programs/cryptography-labs/src/lab1/tests/round-test.c:6 main] Test round 0: key [ securitysecurity ] data [ thisisatestclass ]**

你输入的明文为：thisisatestclass

轮密钥..................

w[0] = 0x73656375 w[1] = 0x72697479 w[2] = 0x73656375 w[3] = 0x72697479

w[4] = 0x8bf7d535 w[5] = 0xf99ea14c w[6] = 0x8afbc239 w[7] = 0xf892b640

w[8] = 0xc6b9dc74 w[9] = 0x3f277d38 w[10] = 0xb5dcbf01 w[11] = 0x4d4e0941

w[12] = 0xedb85f97 w[13] = 0xd29f22af w[14] = 0x67439dae w[15] = 0x2a0d94ef

w[16] = 0x329a8072 w[17] = 0xe005a2dd w[18] = 0x87463f73 w[19] = 0xad4bab9c

w[20] = 0x91f85ee7 w[21] = 0x71fdfc3a w[22] = 0xf6bbc349 w[23] = 0x5bf068d5

w[24] = 0x3dbd5dde w[25] = 0x4c40a1e4 w[26] = 0xbafb62ad w[27] = 0xe10b0a78

w[28] = 0x56dae126 w[29] = 0x1a9a40c2 w[30] = 0xa061226f w[31] = 0x416a2817

w[32] = 0xd4ee11a5 w[33] = 0xce745167 w[34] = 0x6e157308 w[35] = 0x2f7f5b1f

w[36] = 0x1dd7d1b0 w[37] = 0xd3a380d7 w[38] = 0xbdb6f3df w[39] = 0x92c9a8c0

w[40] = 0xf6156bff w[41] = 0x25b6eb28 w[42] = 0x980018f7 w[43] = 0xac9b037

进行AES加密..................

加密完后的密文的ASCII为：

0x3c 0xc 0x2a 0xdb 0x42 0x26 0xb3 0xf 0x3b 0x65 0xab 0x6 0x22 0x10 0x81 0x29

已经将密文写进cryptography0.aes中了,可以在运行该程序的当前目录中找到它。

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:669 deAesFile] 当前解密模式：CBC**

开始解密.........文件名：cryptography0.aes，密文长度：16

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:517 deAes] 当前加密模式：CBC**

解密后的明文ASCII为：

0x74 0x68 0x69 0x73 0x69 0x73 0x61 0x74 0x65 0x73 0x74 0x63 0x6c 0x61 0x73 0x73

明文为：thisisatestclass

现在可以打开cryptography0.aes来查看解密后的密文了！

**[/home/chiro/programs/cryptography-labs/src/lab1/tests/round-test.c:6 main] Test round 1: key [ cryptographylab1 ] data [ liangxinrong200110619liangxinron ]**

你输入的明文为：liangxinrong200110619liangxinron

轮密钥..................

w[0] = 0x63727970 w[1] = 0x746f6772 w[2] = 0x61706879 w[3] = 0x6c616231

w[4] = 0x8dd8be20 w[5] = 0xf9b7d952 w[6] = 0x98c7b12b w[7] = 0xf4a6d31a

w[8] = 0xabbe1c9f w[9] = 0x5209c5cd w[10] = 0xcace74e6 w[11] = 0x3e68a7fc

w[12] = 0xeae2ac2d w[13] = 0xb8eb69e0 w[14] = 0x72251d06 w[15] = 0x4c4dbafa

w[16] = 0x1168104 w[17] = 0xb9fde8e4 w[18] = 0xcbd8f5e2 w[19] = 0x87954f18

w[20] = 0x3b922c13 w[21] = 0x826fc4f7 w[22] = 0x49b73115 w[23] = 0xce227e0d

w[24] = 0x8861fb98 w[25] = 0xa0e3f6f w[26] = 0x43b90e7a w[27] = 0x8d9b7077

w[28] = 0xdc300ec5 w[29] = 0xd63e31aa w[30] = 0x95873fd0 w[31] = 0x181c4fa7

w[32] = 0xc0b45268 w[33] = 0x168a63c2 w[34] = 0x830d5c12 w[35] = 0x9b1113b5

w[36] = 0x59c9877c w[37] = 0x4f43e4be w[38] = 0xcc4eb8ac w[39] = 0x575fab19

w[40] = 0xa0ab5327 w[41] = 0xefe8b799 w[42] = 0x23a60f35 w[43] = 0x74f9a42c

进行AES加密..................

加密完后的密文的ASCII为：

0x6f 0x57 0x26 0x19 0xb8 0x21 0x1a 0x83 0x93 0x57 0xf3 0xd8 0xda 0x42 0xd0 0x3f 0xea 0x8f 0x6c 0x5c 0x9a 0x3c 0x52 0xc6 0xa6 0x3f 0x6d 0x32 0x56 0xff 0xb0 0x8

已经将密文写进cryptography1.aes中了,可以在运行该程序的当前目录中找到它。

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:669 deAesFile] 当前解密模式：CBC**

开始解密.........文件名：cryptography1.aes，密文长度：32

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:517 deAes] 当前加密模式：CBC**

解密后的明文ASCII为：

0x6c 0x69 0x61 0x6e 0x67 0x78 0x69 0x6e 0x72 0x6f 0x6e 0x67 0x32 0x30 0x30 0x31 0x31 0x30 0x36 0x31 0x39 0x6c 0x69 0x61 0x6e 0x67 0x78 0x69 0x6e 0x72 0x6f 0x6e

明文为：liangxinrong200110619liangxinron

现在可以打开cryptography1.aes来查看解密后的密文了！

**[/home/chiro/programs/cryptography-labs/src/lab1/tests/round-test.c:6 main] Test round 2: key [ cryptographylab1 ] data [ liangxinrong200110618liangxinron ]**

你输入的明文为：liangxinrong200110618liangxinron

轮密钥..................

w[0] = 0x63727970 w[1] = 0x746f6772 w[2] = 0x61706879 w[3] = 0x6c616231

w[4] = 0x8dd8be20 w[5] = 0xf9b7d952 w[6] = 0x98c7b12b w[7] = 0xf4a6d31a

w[8] = 0xabbe1c9f w[9] = 0x5209c5cd w[10] = 0xcace74e6 w[11] = 0x3e68a7fc

w[12] = 0xeae2ac2d w[13] = 0xb8eb69e0 w[14] = 0x72251d06 w[15] = 0x4c4dbafa

w[16] = 0x1168104 w[17] = 0xb9fde8e4 w[18] = 0xcbd8f5e2 w[19] = 0x87954f18

w[20] = 0x3b922c13 w[21] = 0x826fc4f7 w[22] = 0x49b73115 w[23] = 0xce227e0d

w[24] = 0x8861fb98 w[25] = 0xa0e3f6f w[26] = 0x43b90e7a w[27] = 0x8d9b7077

w[28] = 0xdc300ec5 w[29] = 0xd63e31aa w[30] = 0x95873fd0 w[31] = 0x181c4fa7

w[32] = 0xc0b45268 w[33] = 0x168a63c2 w[34] = 0x830d5c12 w[35] = 0x9b1113b5

w[36] = 0x59c9877c w[37] = 0x4f43e4be w[38] = 0xcc4eb8ac w[39] = 0x575fab19

w[40] = 0xa0ab5327 w[41] = 0xefe8b799 w[42] = 0x23a60f35 w[43] = 0x74f9a42c

进行AES加密..................

加密完后的密文的ASCII为：

0x6f 0x57 0x26 0x19 0xb8 0x21 0x1a 0x83 0x93 0x57 0xf3 0xd8 0xda 0x42 0xd0 0x3f 0x47 0x92 0xe3 0x8c 0x5 0x89 0x1a 0xf8 0xf5 0xb2 0xb1 0xf1 0x3f 0xfb 0xcf 0xa3

已经将密文写进cryptography2.aes中了,可以在运行该程序的当前目录中找到它。

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:669 deAesFile] 当前解密模式：CBC**

开始解密.........文件名：cryptography2.aes，密文长度：32

**[/home/chiro/programs/cryptography-labs/src/lab1/aes.h:517 deAes] 当前加密模式：CBC**

解密后的明文ASCII为：

0x6c 0x69 0x61 0x6e 0x67 0x78 0x69 0x6e 0x72 0x6f 0x6e 0x67 0x32 0x30 0x30 0x31 0x31 0x30 0x36 0x31 0x38 0x6c 0x69 0x61 0x6e 0x67 0x78 0x69 0x6e 0x72 0x6f 0x6e

明文为：liangxinrong200110618liangxinron

现在可以打开cryptography2.aes来查看解密后的密文了！

1. **实验过程中遇到的问题有哪些？你是怎么解决的。**
2. *问题：程序测试每次都需要重新读取键盘输入数据*

*解决：每次运行的时候配置一个全局配置，忽略键盘输入。*

1. *问题：加密过程中怎么尝试结果都不正确*

*解决：Rcon数组的长度和 PPT 中式子没有对应上，需要给 Rcon[0] 补上*

1. *问题：CBC 模式下输入密文长度有问题*

*解决：修改*int readStrFromFile(char \**fileName*, char \**str*)*，使用 int 而不是 char 承接 getc() 的返回值。*

1. **如果不用lab1-aes.c代码框架或者实现了CBC模式，请说明。**

*本代码框架基于 lab1-aes.c 继续实现，使用 CMake 和 XMake 管理项目，在目录下运行 make 即可得到二进制文件：*

1. *lab1 文件是原始 lab1-aes.c 的功能*
2. *lab1-round-test 文件能够一次测试 ECB、CBC 两个模式下三次加密的输出aes.h 中的 mode 指示了当前运行的模式。*

*在目录下运行 xmake，除了得到上面的二进制文件还会得到 C/Rust 的主程序和单元测试。其中：*

1. *aes-rs 是 Rust 实现的简化版本 AES 加密/解密程序，支持 ECB/CBC 模式。*
2. *lab1-stream-encode、lab1-stream-decode是上述 C 程序的管道模式封装。*

*Rust 在 ECB 模式下使用了异步模式运行，不过运行效率并没有很大的提升。得益于Rust 本身优秀的性能，加密/解密相同大小的数据，Rust用时约为本项目C代码的1/10。*

*运行方法请参考 README.MD，下面放一些截图。*

**➜ chiro@chiro-pc (master) ✗ x c; x b**

**warning: You are working in the project directory(/home/chiro/programs/cryptography-labs) and you can also**

**force to build in current directory via run `xmake -P .`**

**[ 13%]: linking.release aes-rs**

**[ 64%]: cache compiling.release src/lab1/tests/stream-encode.c**

**[ 64%]: cache compiling.release src/lab1/tests/lang-test.c**

**[ 64%]: cache compiling.release src/lab1/tests/function-test.c**

**[ 64%]: cache compiling.release src/lab1/lab1-aes.c**

**[ 64%]: cache compiling.release src/lab1/tests/stream-decode.c**

**[ 64%]: cache compiling.release src/lab1/tests/round-test.c**

**[ 76%]: linking.release lab1-stream-encode**

**[ 76%]: linking.release lab1-lang-test**

**[ 76%]: linking.release lab1-function-test**

**[ 76%]: linking.release lab1**

**[ 76%]: linking.release lab1-stream-decode**

**[ 76%]: linking.release lab1-round-test**

**[100%]: build ok!**

**warning: You are working in the project directory(/home/chiro/programs/cryptography-labs) and you can also**

**force to build in current directory via run `xmake -P .`**

**➜ chiro@chiro-pc (master) ✗ file aes-rs**

**aes-rs: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=f67c4a003fa0e2893695b72910f91e0f4d6f2944, for GNU/Linux 4.4.0, with debug\_info, not stripped**

**➜ chiro@chiro-pc (master) ✗ cat aes-rs | ./lab1-stream-encode 1145141919810aaa CBC > aes-rs-encoded**

**➜ chiro@chiro-pc (master) ✗ cat aes-rs-encoded | ./lab1-stream-decode 1145141919810aaa CBC > aes-rs-decoded**

**➜ chiro@chiro-pc (master) ✗ file aes-rs-decoded**

**aes-rs-decoded: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=f67c4a003fa0e2893695b72910f91e0f4d6f2944, for GNU/Linux 4.4.0, with de**

**bug\_info, not stripped**

**➜ chiro@chiro-pc (master) ✗ ./aes-rs -i aes-rs -o aes-rs-encoded-2**

**11:46:05 [INFO] args: input=aes-rs, output=aes-rs-encoded-2, direction=encode, mode=ECB, key=1145141919810aaa**

**➜ chiro@chiro-pc (master) ✗ ./aes-rs -i aes-rs-encoded-2 -o aes-rs-decoded-2 -d decode**

**11:47:43 [INFO] args: input=aes-rs-encoded-2, output=aes-rs-decoded-2, direction=decode, mode=ECB, key=1145141919810aaa**

**➜ chiro@chiro-pc (master) ✗ file aes-rs-decoded-2**

**aes-rs-decoded-2: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=f67c4a003fa0e2893695b72910f91e0f4d6f2944, for GNU/Linux 4.4.0, with**

**debug\_info, not stripped**

**➜ chiro@chiro-pc (master) ✗**

**➜ chiro@chiro-pc ~/programs/cryptography-labs/build/linux/x86\_64/release git:(master) ✗ ./aes-rs --help**

**USAGE:**

**aes-rs [OPTIONS]**

**OPTIONS:**

**-d, --direction Decode or encode data [default: encode] [possible values: decode,**

**encode, both]**

**-h, --help Print help information**

**-i, --input Input filename [default: stdin]**

**-k, --key Decode / encode key [default: 1145141919810aaa]**

**-m, --mode Run mode [default: ECB] [possible values: ECB, CBC]**

**-o, --output Output filename [default: stdout]**