МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра информационной безопасности

ОТЧЕТ

по лабораторной работе №4-5
по дисциплине «Криптографические методы защиты информации»
Тема: Изучение шифра DES и AES

Студентка гр. 9363 _____ Труханова В.А.

Преподаватель _____ Племянников А.К.

Санкт-Петербург
2023

Цель работы

Исследовать шифры DES, 3DES, а также другие модификациии шифра DES: DESX, DESL, DESXL и получить практические навыки работы с ними, в том числе с использованием приложения Cryptool 1 и 2.

Исследовать характеристики шифра Rijndael и других финалистов конкурса AES, а также изучить атаку предсказанием дополнения на симметричные блочные шифры в режиме использования CBC. Получить практические навыки работы с шифрами и алгоритмом проведения атаки, в том числе с использованием приложения CrypTool 1 и 2.

4.1. Шифр DES

Задание

1. Изучить преобразования шифра DES с помощью демонстрационного приложения из CrypTool 1: Indiv.Procedures –> Visualization… –> DES…

4.1.1. Основные характеристики и описание DES

Шифр DES - блочный шифр с симметричными ключами. Открытый текст шифруется блоками 64 бит, используя 64 битный ключ шифра (56 битов фактический ключ + 8 битов четности). На рисунке 1 представлена общая структура шифрования DES.

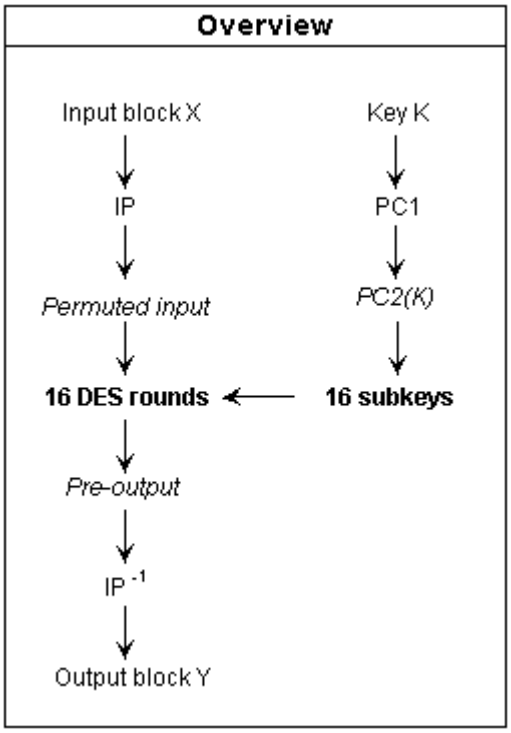


Рисунок 1 – Общая схема шифрования

Исходный текст делится на блоки по 64 бита. На вход шифрования подается один блок и 64-битный ключ. Шифрование состоит из двух перестановок (в начале и в конце) и 16-ти раундов Фейстеля.

1 шаг состоит в перестановке битов входного блока согласно таблице перестановки. Пример данной перестановки представлен на рисунке 2.

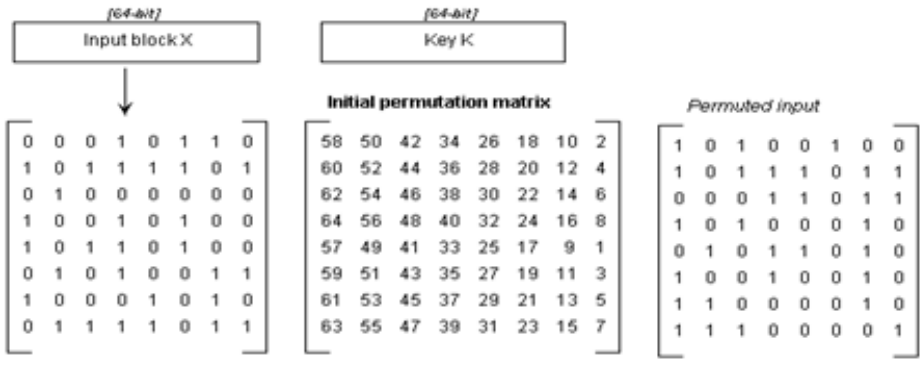


Рисунок 2 – Перестановка битов входного блока

Затем происходит формирование раундовых ключей. Сначала из исходного 64-битного ключа удаляются биты проверки, стоящие на местах «8,16,24,32,40,48,56,64». Затем в соответствии с матрицей перестановки битов получаем 56-битный ключ шифрование, получение которого изображено на рисунке 3.

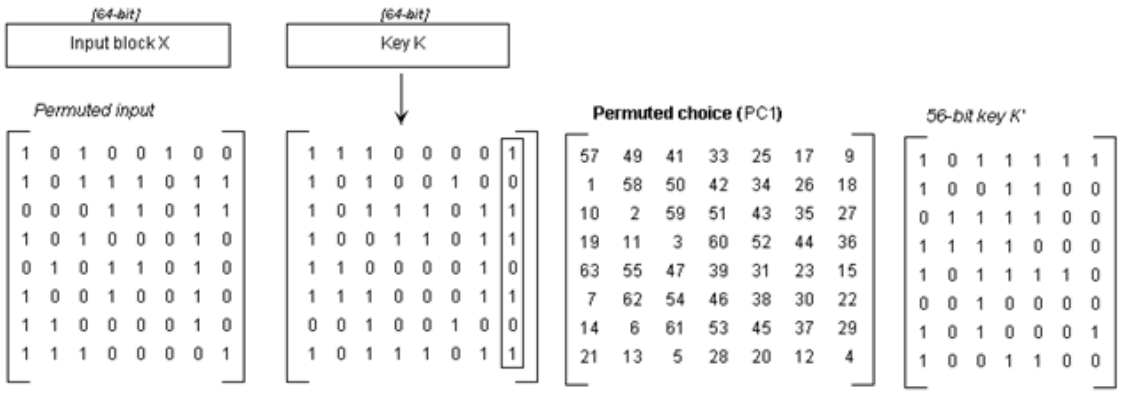


Рисунок 3 – Получение 56-битного ключа шифрования

Затем полученный ключ делится на два блока по 28 бит – правый(R) и левый(L) и каждая часть сдвигается на определенное количество бит в зависимости от раунда. В раундах «1,2,9,16» сдвиг происходит на 1 бит, в остальных раундах на 2 бита. После сдвига части опять соединяются и в соответствии с матрицей перестановки бит сжимаются в 48-битный ключ. Пример представлен на рисунке 4.

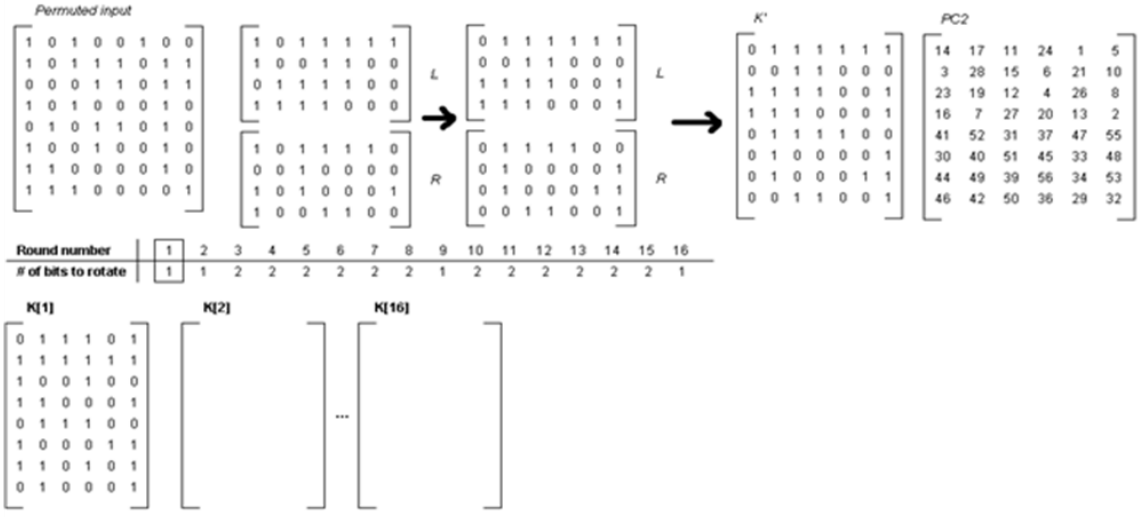


Рисунок 4 – Получение 48-битного ключа

Следующие раундовые ключи формируются на основе предыдущих субблоков правого и левого все 16 раундов.

Главный блок шифрования основан на схеме Фейстеля, повторяющейся 16 раз, выполняя 16 раундов. В каждом раунде используется свой субключ полученный в предыдущих шагах. Входной 64-битный блок делится на два блока по 32-бита – правый(R) и левый(L). Следующее значение левого блока равно предыдущему значению правого блока, а следующее

значение правого блока равно предыдущему значению правого блока, зашифрованного функцией шифрования с раундовым ключом и проксоренным полученным значением с левым ключом. Пример одного раунда представлен на рисунке 5.

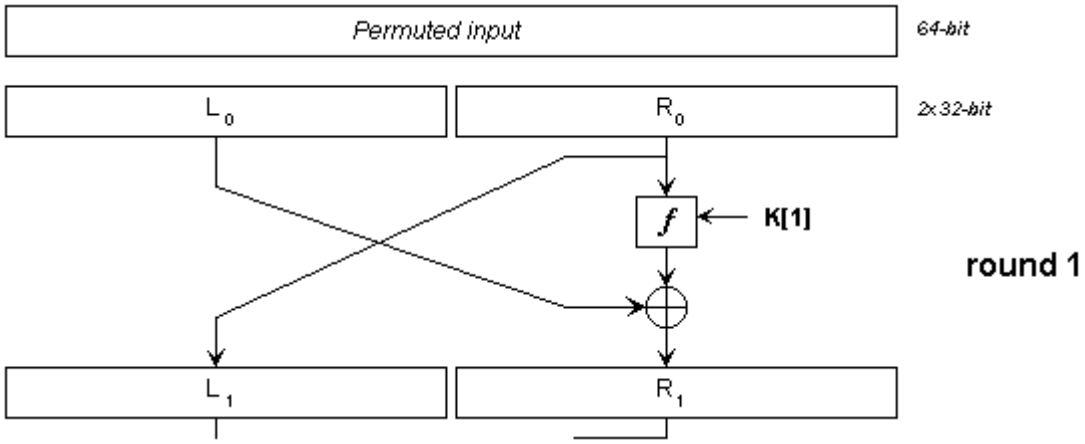


Рисунок 5 – Один раунд шифрования блока текста

Рассмотрим функцию шифрования. На вход поступает 32-битный блок, который расширяется до 48 бит по правилу: значение последнего бита добавляется в начало блока, значение перового бита добавляется в конец блока, а значения битов «4,5», «8,9», «16,17», «24,25», «32,33», «40,41», «48,49» дублируются меняются местами и встают между соответствующими битами. Схема расширения представлена на рисунке 6.

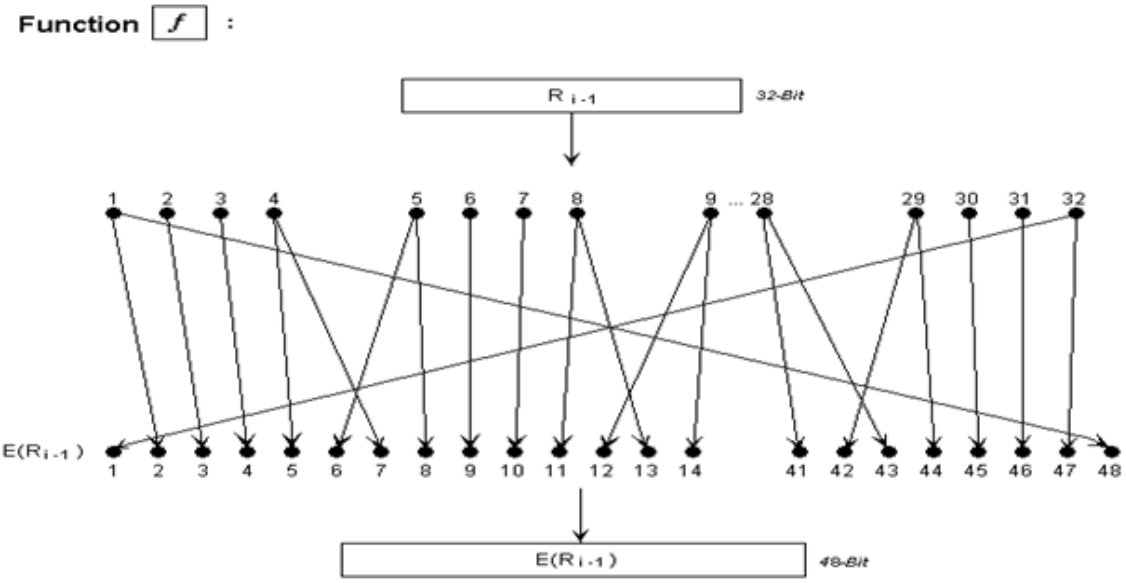


Рисунок 6 – Схема расширения 32-битного блока

Затем полученный блок ксорится с раундовым ключом. Схема работы представлена на рисунке 7. Полученный блок делится на 8 S-блоков, каждому из которых соответствует таблица. Комбинация битов 1 и 6 на входе определяет одну из четырех строк Комбинация битов от 2 -го до 5 -го определяет один из шестнадцати столбцов 4-х битовая подстановка берется из клетки на пересечении строки и столбца. Пример работы представлен на рисунке 8.

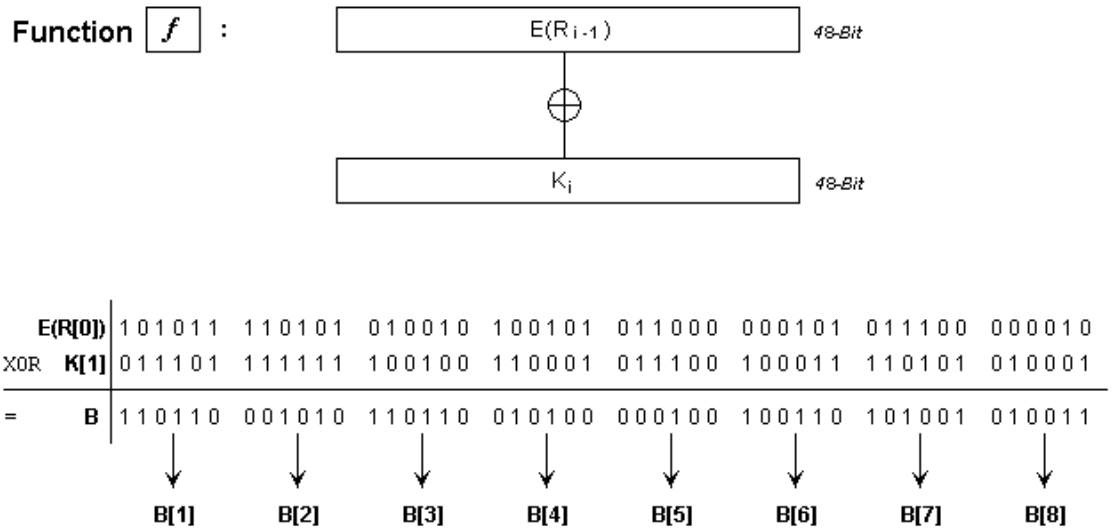


Рисунок 7 – Операция XOR

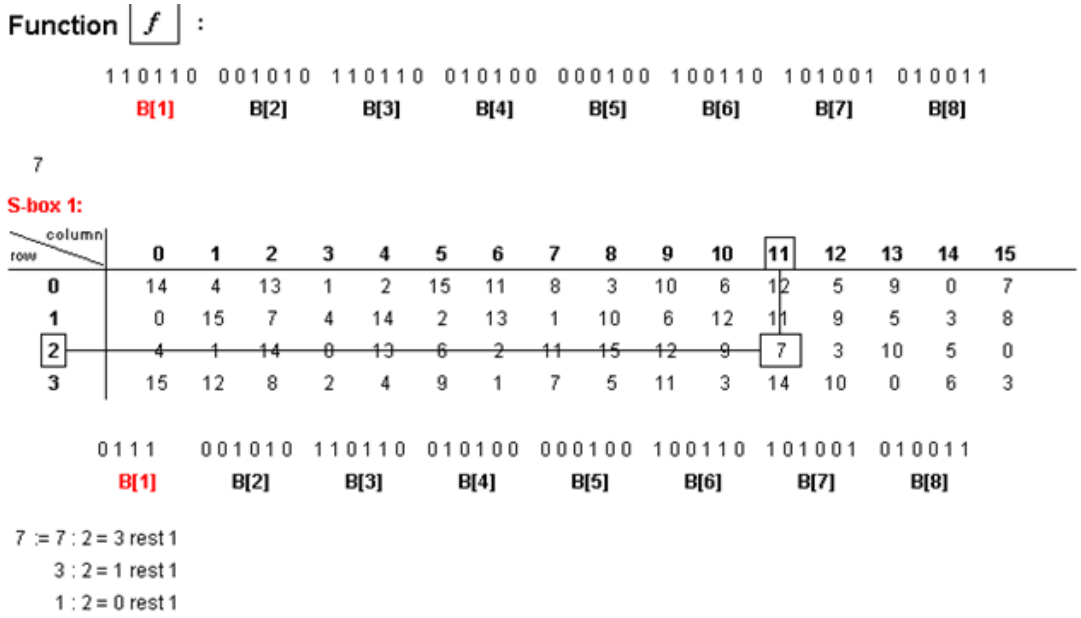


Рисунок 8 – Получение 32-битного блока

Затем полученный блок переставляется в соответствии с матрицей перестановки битов. Пример представлен на рисунке 9.

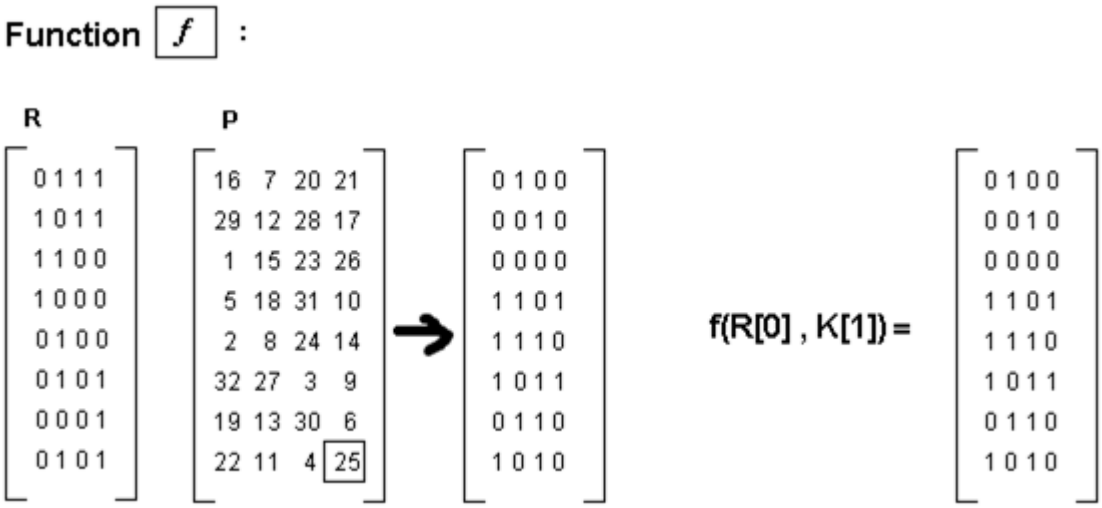


Рисунок 9 – Перестановка битов блока

После 16ти раундов полученный 64-битный блок переставляется в соответствии с матрицей перестановки битов. Работа представлена на рисунке 1.10. Работа шифрования блока завершена.

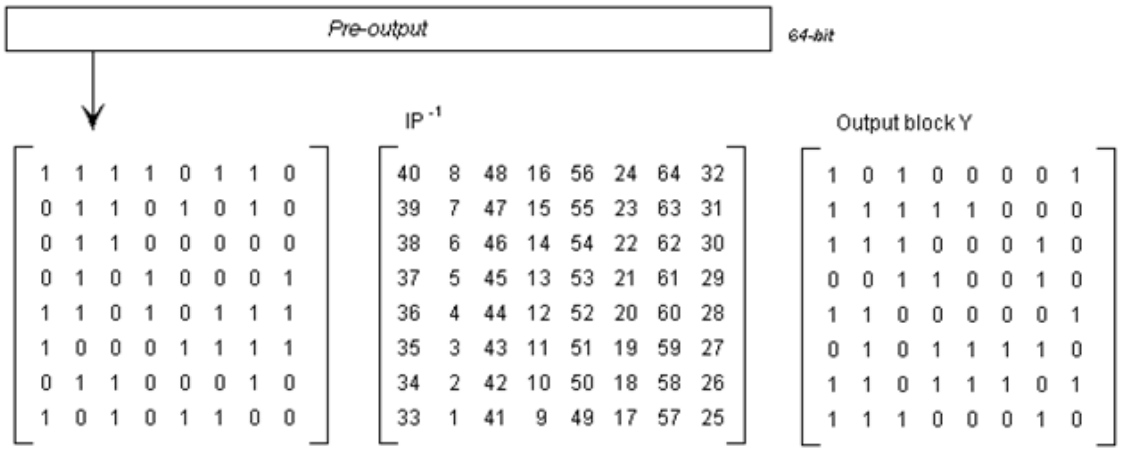


Рисунок 10 – Перестановка битов блока перед выводом шифрблока

4.2. Исследование DES в режимах ECB и CBC

Задание

1. Создать картинку со своими ФИО (формат bmp);
2. Зашифровать картинку шифром DES в режиме ECB;
3. Зашифровать картинку шифром DES в режиме CBC с тем же ключом;
4. Сохранить скриншоты картинок для отчета;

5. Сжать исходную и 2 зашифрованных картинки средствами CrypTool; Зафиксировать размеры полученных файлов в таблице.

4.2.1. Схемы использования DES в режимах ECB и CBC

В режиме ECB шифра DES используется независимо для каждого 64-битного блока шифруемых данных. Схема использования шифра в режиме ECB представлена на рисунке 11.

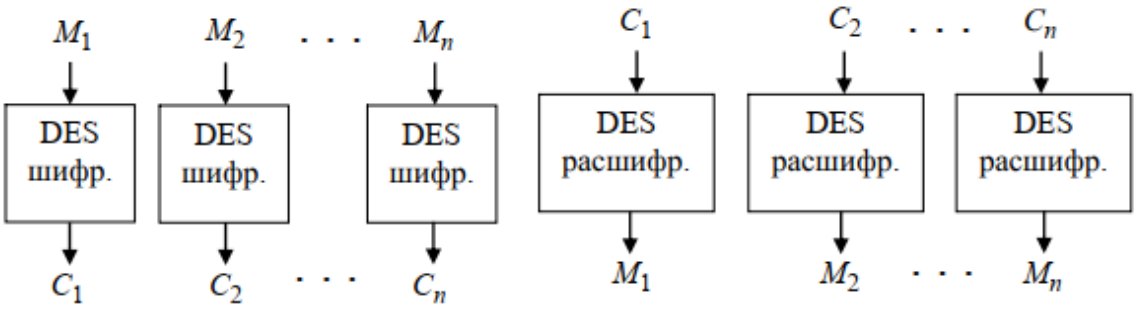


Рисунок 11 – Схема режима ECB

В режиме CBC перед запуском DES для зашифрования каждого очередного блока открытого текста происходит побитовое XOR-сложение этого блока с блоком зашифрованного текста из предыдущего шага, а первый блок побитово XOR-складывается с вектором инициализации (IV):

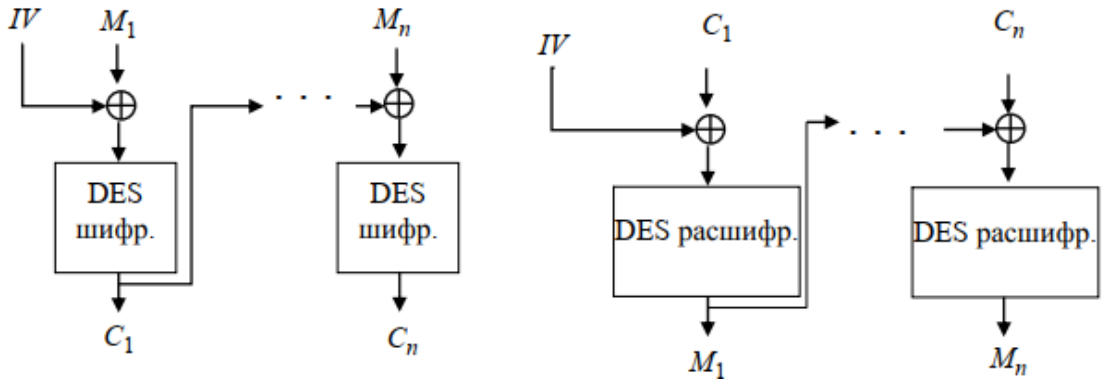


Рисунок 12 – Схема режима CBC

4.2.2. Скриншоты исходного и шифрованных изображений в разных режимах работы шифров

Создаем картинку со своими ФИО (формат bmp). Исходная картинка представлена на рисунке 13.

Труханова
Вероника
Александровна

Рисунок 13 – Файл имя.BMP

Зашифровываем картинку шифром DES в режиме ECB.

Ключ шифра: “01 02 03 04 05 06 07 08”. Результат шифровки представлен на рисунке 14:

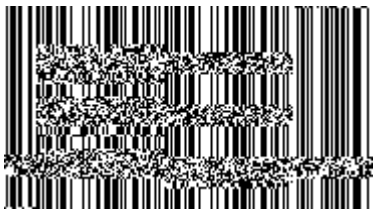


Рисунок14 – Файл DES-ECB-имя.bmp

Зашифруем картинку шифром DES в режиме CBC с тем же ключом.

Результат шифровки представлен на рисунке 15:

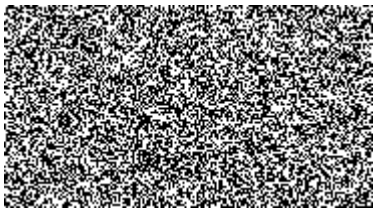


Рисунок 15 – Файл DES-CBC-имя.bmp

Сжать исходную и 2 зашифрованных картинки средствами CrypTool. Зафиксировать размеры полученных файлов в таблице.

Название файла	имя.bmp	DES-ECB-имя.bmp	DES-CBC-имя.bmp
До сжатия, байт	2 486	2 486	2 488
После сжатия, байт	533	972	2 488

4.3. Исследование 3-DES

Задание

1.Определить экспериментальным путем по какой схеме работает реализация 3-DES в CrypTool. Сохранить подтверждающие скриншоты.

4.3.1. Основные параметры и обобщенная схема шифра

Шифр 3-DES (рисунок 16) это трехкратное применение обычного DES:

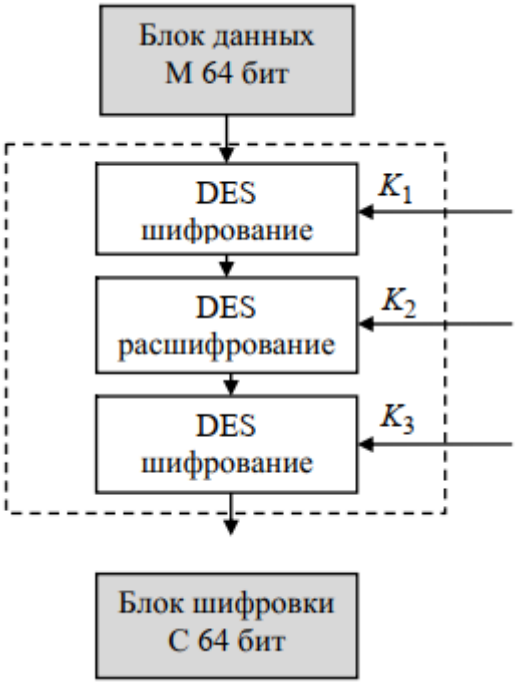


Рисунок 16 – Схема шифра 3-DES

Существует 4 основные версии данного шифра:

1. DES-EEE3 – шифрование происходит 3 раза независимыми ключами.

2. DES-EDE3 – операции шифровка-расшифровка-шифровка с тремя разными ключами.

3. DES-EEE2 – то же что и DESEEE3, но на первом и последнем шаге одинаковый ключ.

4. DES-EDE2 – то же что и DESEDE3, но на первом и последнем шаге используется один и тот же ключ.

На текущий момент самыми популярными версиями шифра являются DES-EDE3 и DES-EDE2.

4.3.2. Проведение эксперимента

Был выбран текст длиной не менее 1000 символов (Рисунок 17).

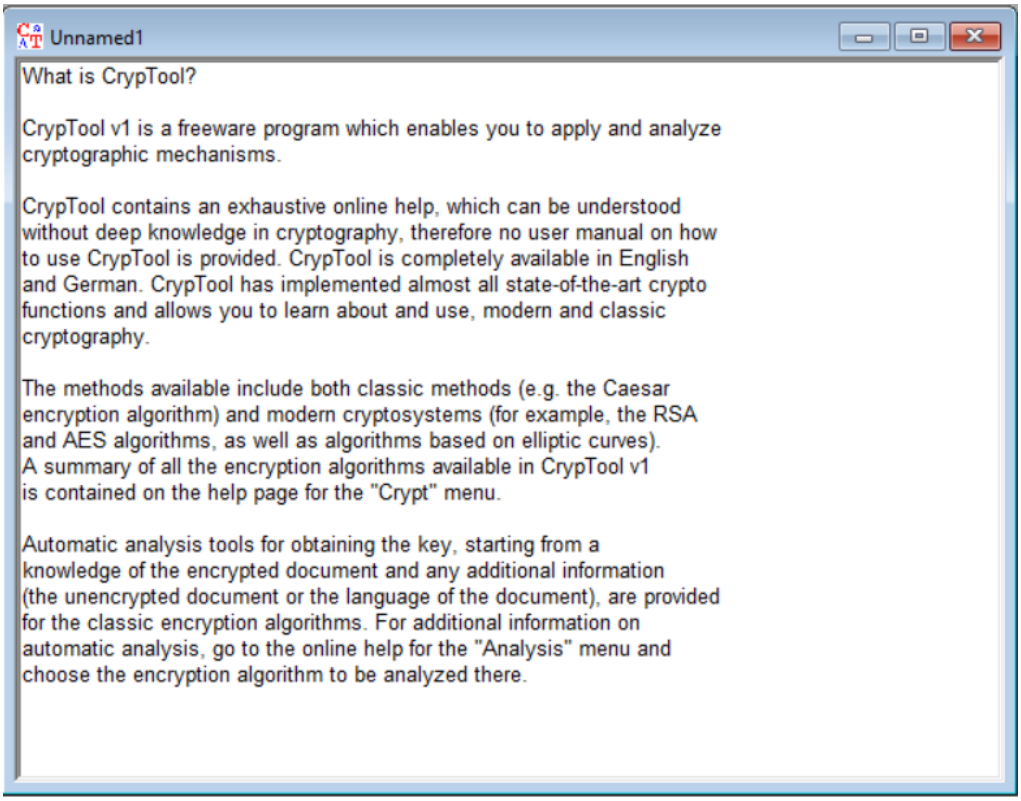


Рисунок 17 – Выбранный текст длиной 1249 символов

Зашифруем файл с ключом «11 40 23 17 33 66 16 36» шифрам DES (ECB) и с ключом «11 40 23 17 33 66 16 36 11 40 23 17 33 66 16 36» 3-DES (ECB) и сравним полученных результатов (рисунок 18 и рисунок 19 соответственно):

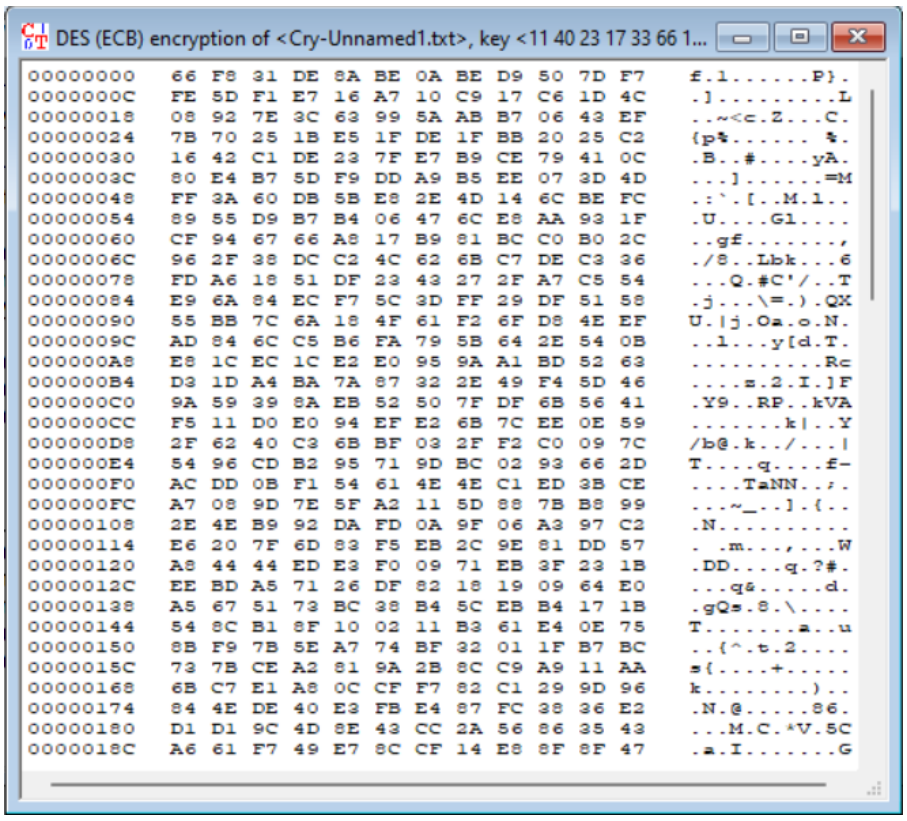


Рисунок 18 – Результат работы DES

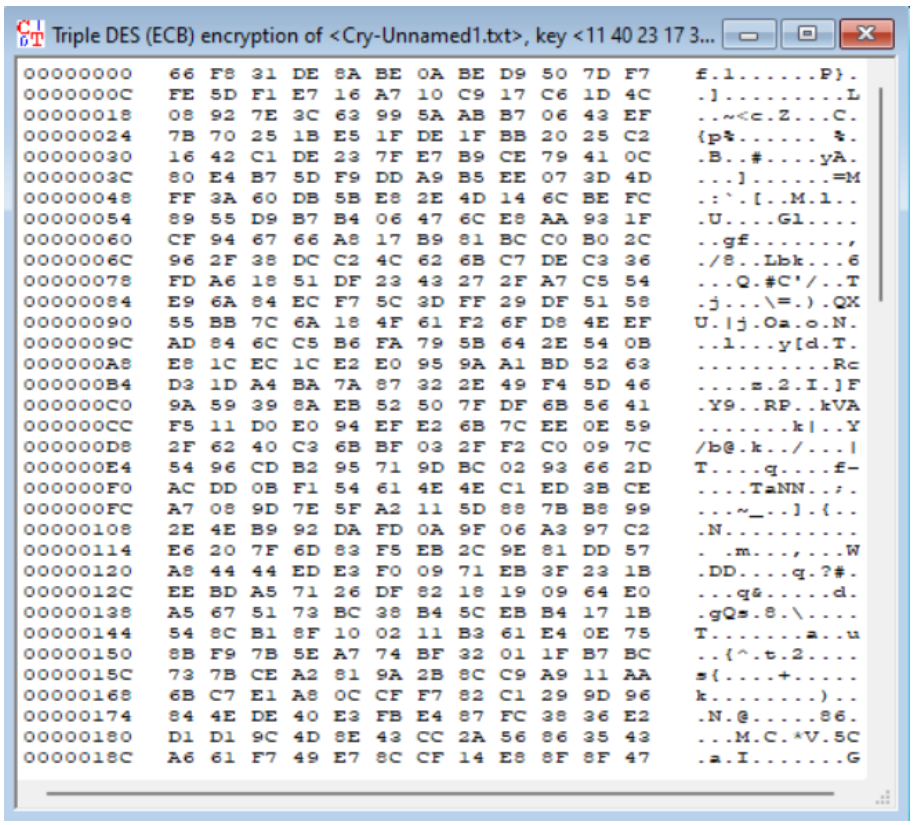


Рисунок 19 – Результат работы 3-DES

Как видно по рисункам результаты полностью совпали, на основе этого мы можем сделать вывод, что 3-DES шифрование имеет вид EDE2 или EDE3. Однако, так как ключ шифрования 3-DES составляет 128 бит, можно сделать вывод, что на двух из трех этапах шифрования используется один и тот же ключ. Таким образом получается, что в CrypTool производится шифрование DES-EDE2 $C = E_{k_1}(E_{k_2}^{-1}(E_{k_1}(P)))$

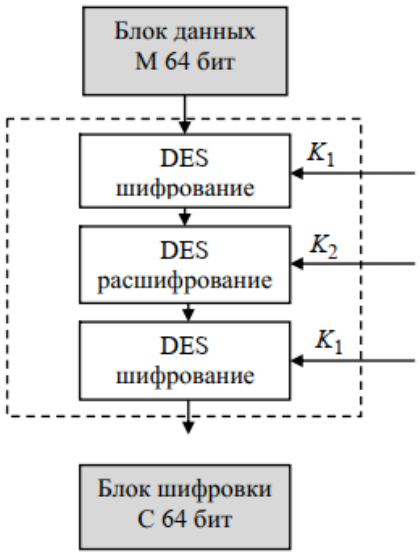


Рисунок 20 – Схема реализации в 3-DES в CrypTool 1

5.1. Шифр AES

Задание

1. Изучить преобразования шифра AES с помощью демонстрационного приложения из CrypTool 1: Indiv.Procedures –> Visualization… –> AES –> Rijndael Animation;
2. Провести наблюдения в потоковой модели шифра AES с помощью демонстрационного приложения из CrypTool 1 для 0-текста и 0-ключа: Indiv.Procedures –> Visualization… –> AES –> Rijndael Flow Visualisation.

5.1.1. Исследование преобразований AES

Шифр AES (в прежнем Rijndael) работает на основе перестановочноподстановочной сети (SP-сеть). Обобщенная схема работы

алгоритма представлена на рисунке 22. Шифр принимает на вход блок данных 128 бит и ключ с вариантами длиной 128, 192 и 256 бит, выполняя раундовое преобразование 10, 12 и 14 раз соответственно.

В версии с наименьшей длиной ключа алгоритм AES получает на вход блок открытого текста размером 128 бит (16 байт) и ключ того же размера. Значения блока записываются в столбцы матрицы состояний размером 4 × 4 байт.

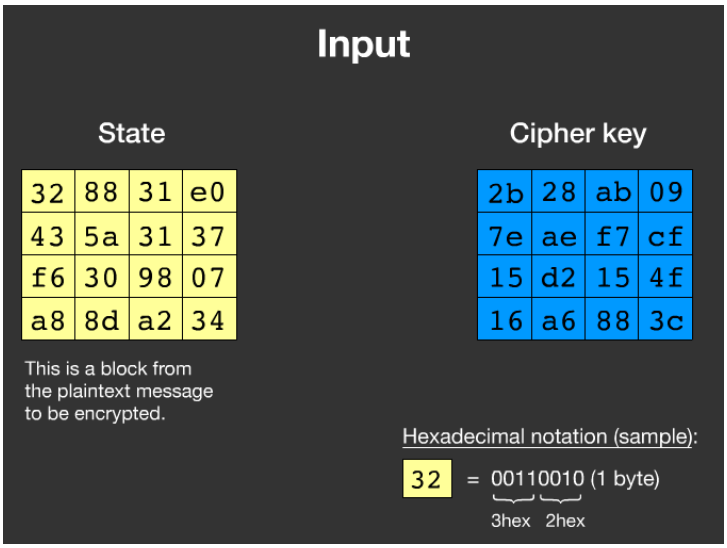


Рисунок 21 – Пример входного блока и пример ключа

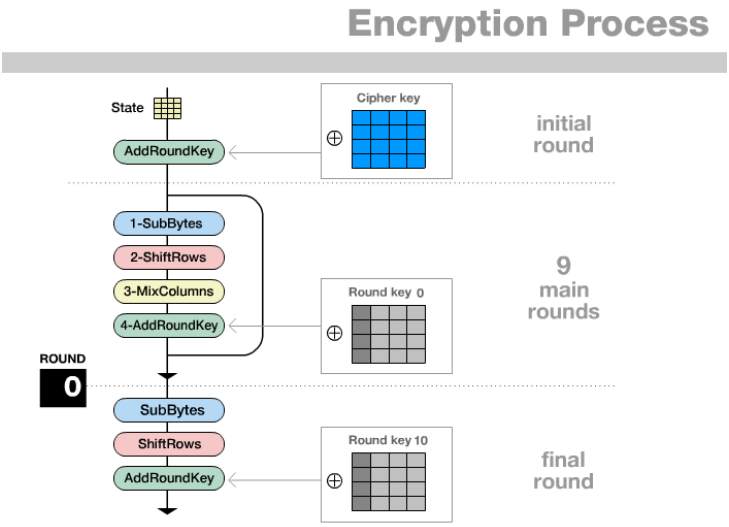


Рисунок 22 – Схема реализации шифра AES

Процедура расширения ключей ExpandKey создает последовательно (слово за словом) 128-битные раундовые ключи N_r раундов от единственного входного ключа шифра.

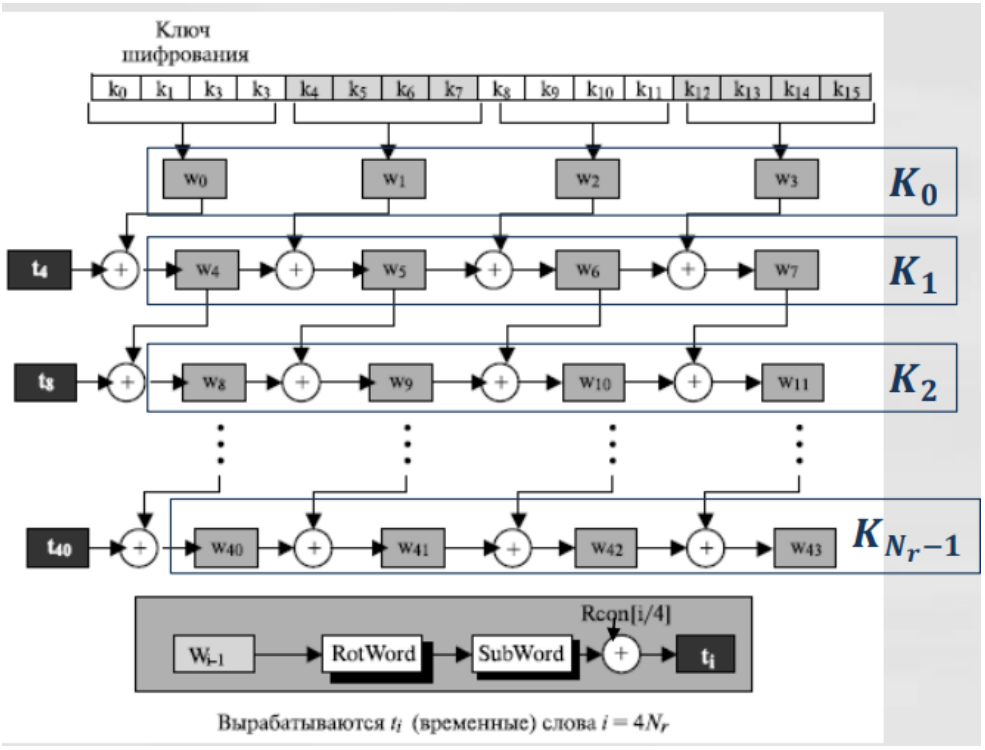
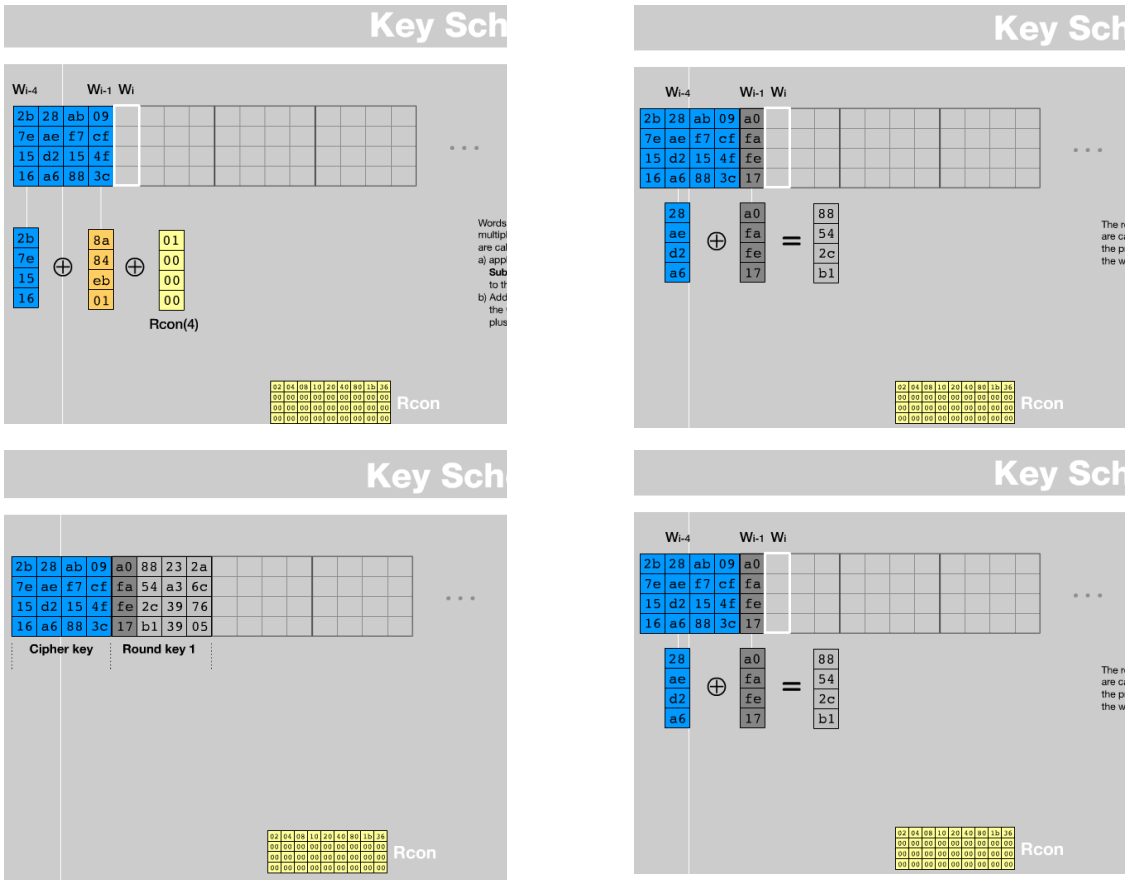


Рисунок 23 – Пример генерации раундовых ключей для AES-128



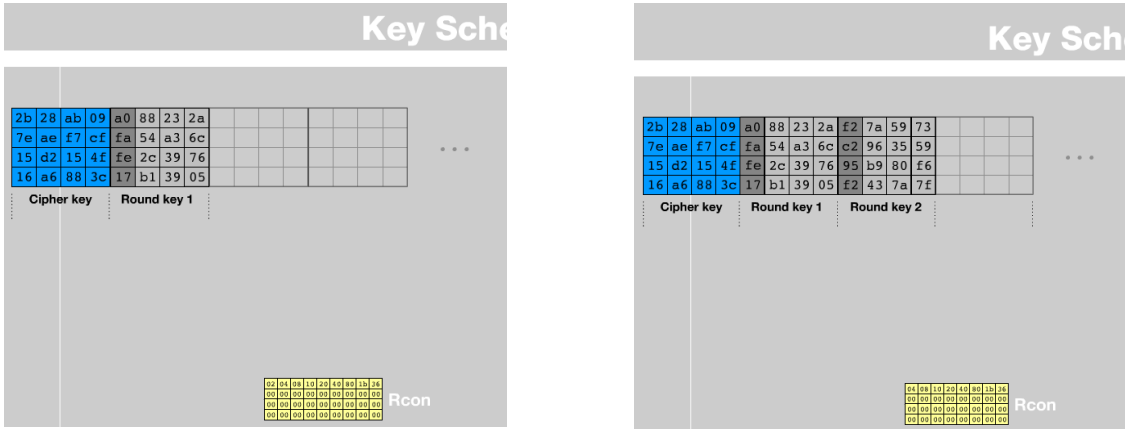


Рисунок 24 – Пример последовательности шагов генерации раундовых ключей для AES-128

После того как сформированы раундовые ключи, начинается раундовая обработка матрицы состояний. В каждом раунде алгоритма выполняются следующие преобразования, представленные на рис. 25:

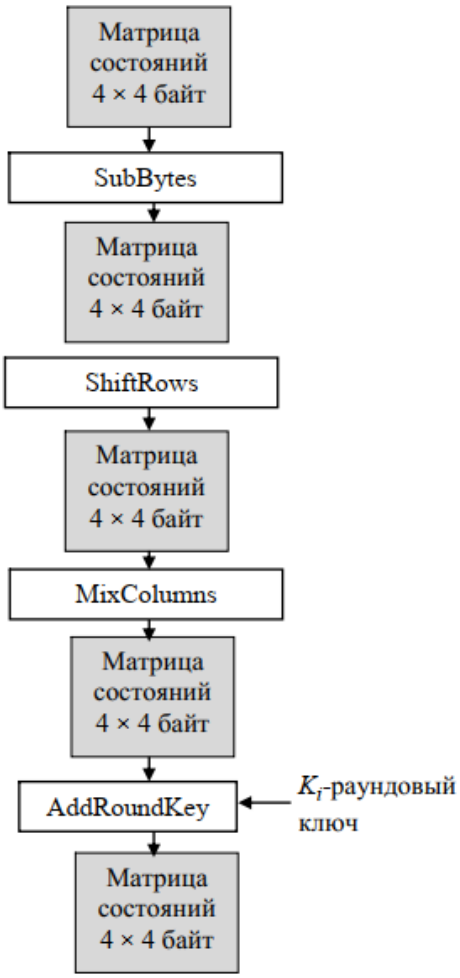


Рисунок 25 – Последовательность преобразования в каждом раунде

- Столбцы матрицы состояний складываются с ключом шифра операцией XOR.
- Полученная матрица состояний проходит через преобразование подстановки SubBytes.
- Циклический сдвиг влево всех строк матрицы состояний выполняется преобразованием ShiftRows.
- Смешивание столбцов матрицы состояний путем ее умножением на матрицу констант в конечном поле *GF*(28) выполняется преобразованием MixColumn.
- Сложение полученных столбцов матрицы состояний с раундовым ключом операцией xor – преобразование AddRoundKey.

Действия 2–5 повторяются в каждом раунде, за исключением последнего. Последний раунд не включает в себя смешивание столбцов.

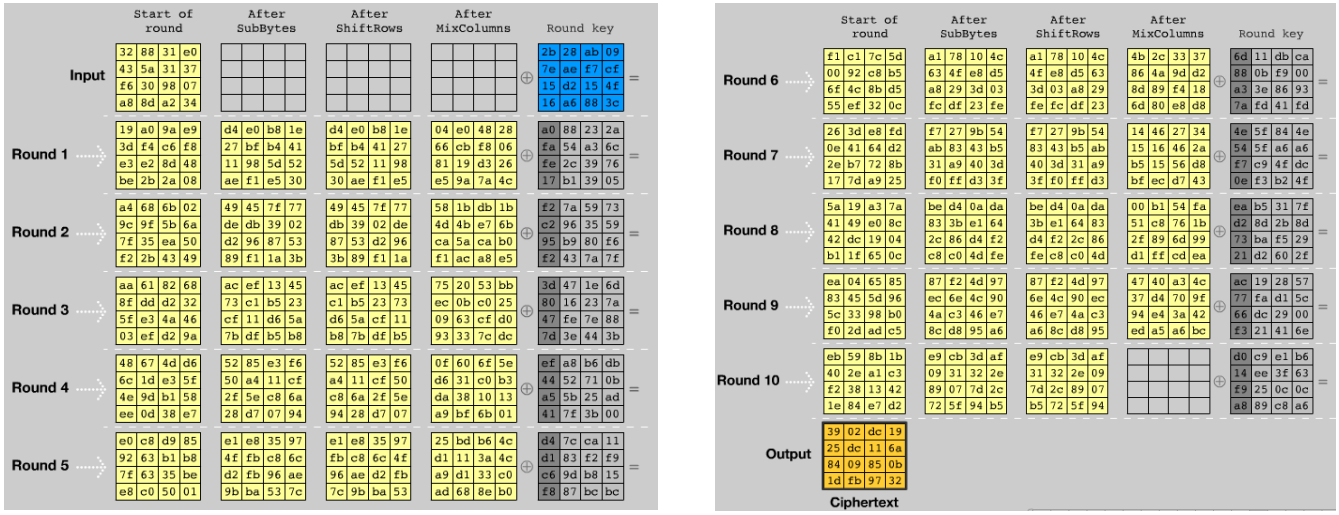


Рисунок 26 – Пример реализации шифра AES

1. Скриншоты наблюдений потоковой модели шифра и сопутствующие выводы

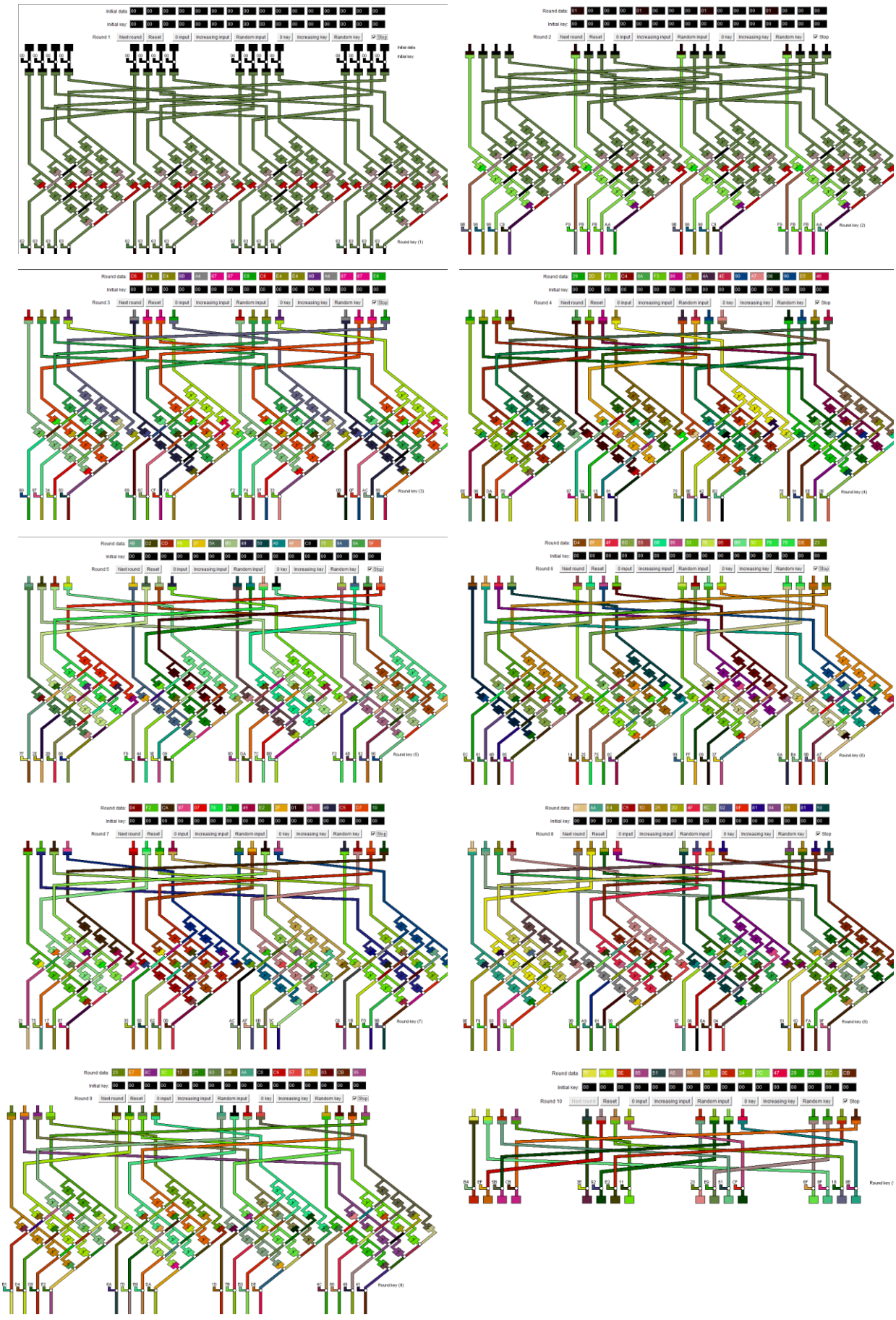


Рисунок 27 – Потоковая модель шифра AES и сопутствующие выводы

6. Атака предсказанием дополнения на шифр AES в режиме CBC (Padding Oracle Attack)

Задание

1. Найти и запустить шаблон атаки в CrypTool 2: Padding Oracle Attack on AES;
2. Подготовьтесь к атаке теоретически:
   - Изучите комментарии к шаблону;
   - Изучите публикацию.
3. Внедрите во второй блок исходного текста коды символов своего имени;
4. Выполните 3 фазы атаки и сохраните итоговые скриншоты по окончанию каждой фазы;
5. Убедитесь, что атака удалась.

6.1. Исходные данные

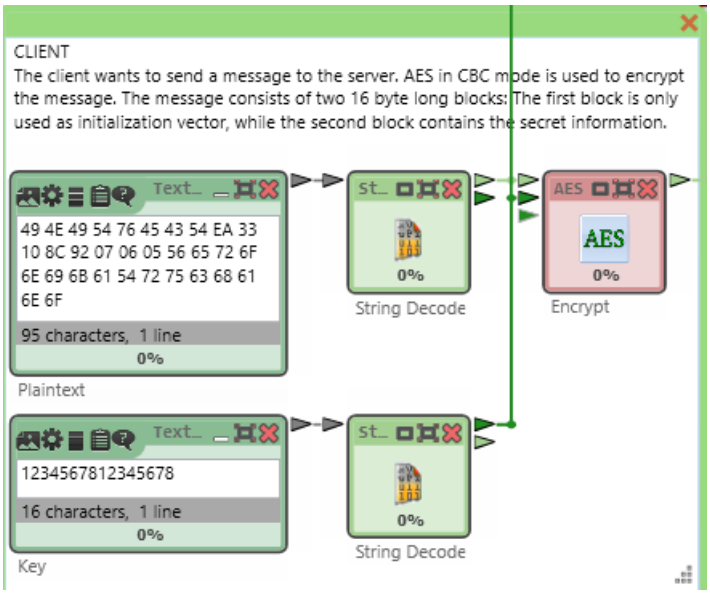


Рисунок 28 – Исходные данные

Исходное сообщение: VeronikaTrukhano

Бинарный вид открытого текста:

56 65 72 6F 6E 69 6B 61 54 72 75 63 68 61 6E 6F

Ключ: 1234567812345678

6.2. Шаблон атаки «Padding Oracle Attack» из CrypTool 2

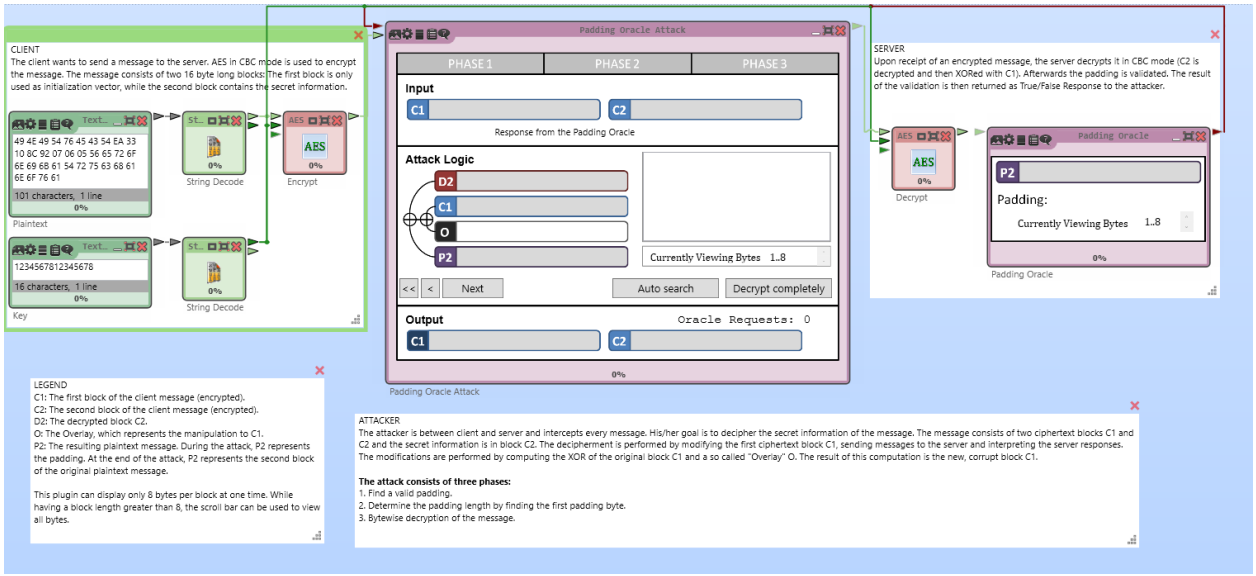


Рисунок 29 – Шаблон атаки

C1: первый блок клиентского сообщения (зашифрованный).

C2: второй блок клиентского сообщения (зашифрованный).

D2: расшифрованный блок C2.

O: наложение, представляющее манипуляцию с C1.

P2: результирующее текстовое сообщение. Во время атаки P2 представляет собой отступ. В конце атаки, P2 представляет второй блок исходного текста сообщения.

6.3. Описание атаки «Padding Oracle Attack»

Client хочет отправить сообщение на сервер. AES в режиме CBC используется для шифрования сообщения. Сообщение состоит из двух блоков длиной 16 байт: первый блок используется в качестве вектора инициализации, а второй блок содержит секретную информацию. После получения зашифрованного сообщения сервер расшифровывает его в режиме CBC (C2 расшифровывается и затем xor с C1). После заполнения проверяется. Результат проверки возвращается злоумышленнику в виде ответа True/False.

Злоумышленник находится между клиентом и сервером и перехватывает каждое сообщение. Его цель - расшифровать секретную информацию из сообщения. Сообщение состоит из двух блоков шифротекста C1 и C2 и секретной информации в блоке C2. Расшифровка выполняется путем изменения первого блока шифротекста C1, отправки сообщений на сервер и

интерпретации ответов сервера. Изменения выполняются путем вычисления XOR исходного блока C1 и так называемого "наложения" O. Результатом этого вычисления является новый, поврежденный блок C1.

6.4. Результаты фаз атак

Первая фаза: Нахождение длины дополнения, т.е. последний байт

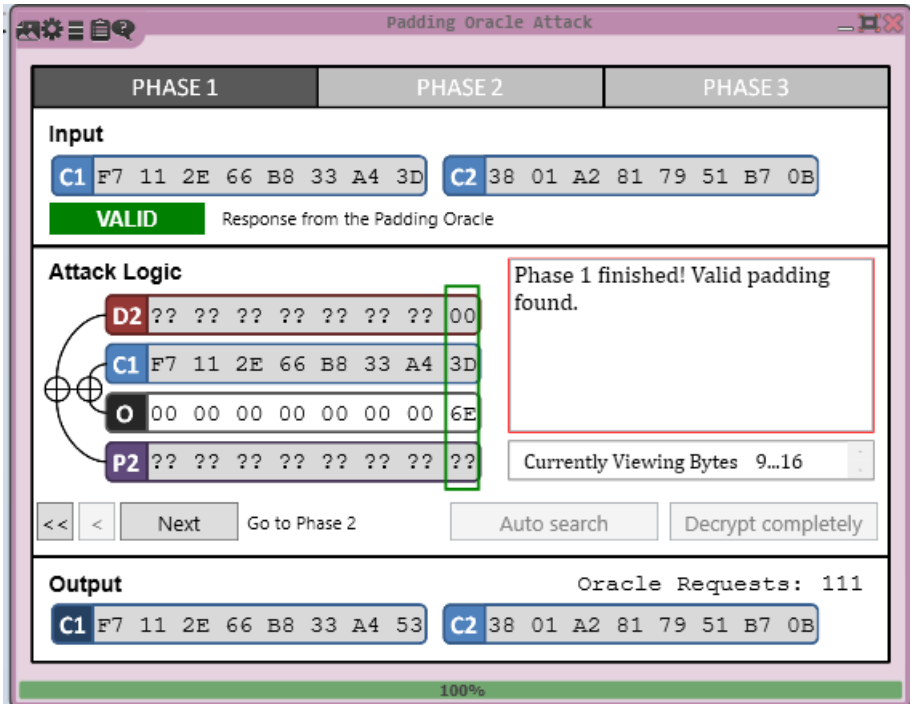


Рисунок 30 – Фаза 1

Вторая фаза: Подбор дополнения

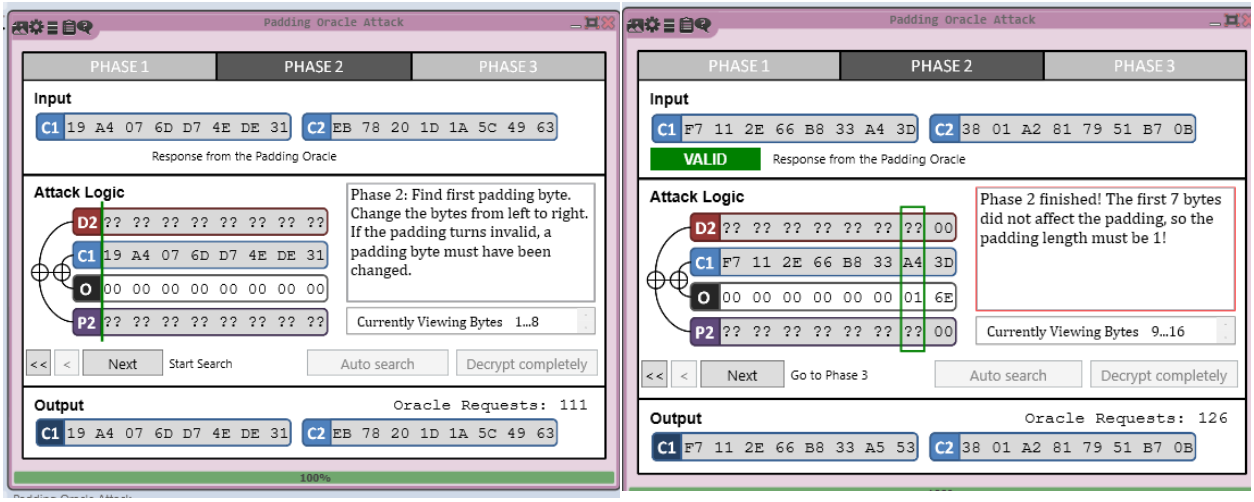


Рисунок 31 – Фаза 2

Третья фаза: Расшифровка текста

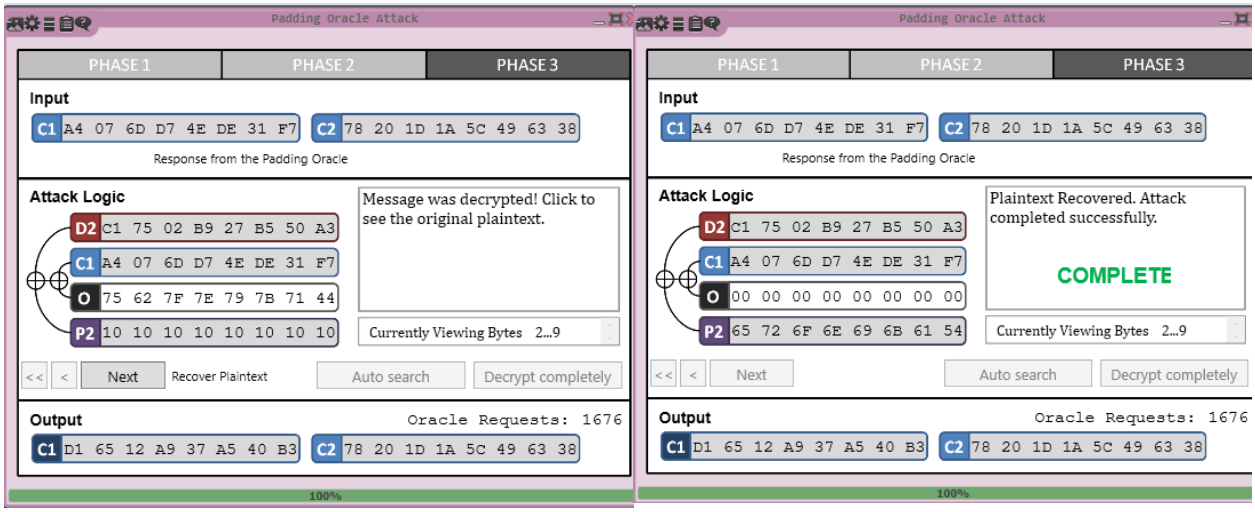


Рисунок 3 – Успешное завершение фазы 3

Заключение

Шифр	DES
Тип шифра	Комбинированный, блочный (64 битов)
Тип ключа	Битный ключ
Длина ключа	64 битов (56 битов фактический ключ + 8 битов четности)
Количество раундов	16

Шифр	DES-EEE3	DES-EDE3	DES-EEE2	DES-EDE2
Схема	Шифрование происходит 3 раза	Шифровка-расшифровка-шифровка	Шифрование происходит 3 раза	Шифровка-расшифровка -шифровка
Ключ	192 бит, 3 разных ключа	192 бит, 3 разных ключа	128 бит, K1 = K3	128 бит, K1 = K3

Шифр	AES		
Тип шифра	Комбинированный, блочный		
Тип ключа	Битный ключ		
Длина блока	128 бит	128 бит	128 бит
Длина ключа	128 бит	192 бит	256 бит
Количество раундов	10	12	14

- Зная правило дополнения блока открытого текста, шифр AES в режиме CBC уязвим к Padding Oracle Attack. Padding Oracle Attack – атака, не требуемая вычисления исходного ключа, позволяющая найти исходный текст за счет перехвата шифрблоков, поступающих на сервер проверки корректности дополнения.