ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ СИСТЕМЫ ШИФРОВАНИЯ НА ОСНОВЕ ОДНОРАЗОВЫХ БЛОКНОТОВ

Щебетов Андрей Сергеевич

11 класс ЧОУ СОШ «Ломоносовская школа» (г. Москва)

Презентация на конференции «Интел-Авангард»

23 февраля 2018 года

ЦЕЛИ И ЗАДАЧИ ПРОЕКТА

- *Практическая **реализация работоспособного прототипа** системы шифрования, использующей одноразовые блокноты*
- ***Сочетание шифрования одноразовыми блокнотами** с кодовыми таблицами и супершифрованием*
- ***Практическая проверка 3 предложенных протоколов** обмена кодовыми таблицами, одноразовыми блокнотами и сообщениями*
- ***Понимание из каких блоков и элементов и на какой программной и аппаратной базе** может быть реализован такой прототип*
- ***Проверка пяти ГСЧ** для генерации кодовых таблиц и одноразовых блокнотов*
- *Основной приоритет – **обеспечение информационной, программной и аппаратной безопасности***
- ***Простота и наглядность** созданного прототипа и возможности прямой проверки правильности работы системы на каждом шаге*
- ***Обеспечение физической безопасности системы** и наличие «достоверного отрицания»*
- ***Намеренное упрощение:***
  - *Наглядные **арифметические операции** взятия по модулю (вместо побитового XOR)*
  - ***Собственный алфавит** из 256 символов (вместо 55295 несуррогатных символов UTF-16)*
  - ***Уменьшенная степень автоматизации** (ручная отправка ZIP-контейнеров из почтового клиента)*
  - ***Консольная реализация***

Задачей проекта являлась построение прототипа системы обмена сообщениями с помощью одноразового блокнота, кодовых таблиц и супершифрования

ШЕСТЬ ПРИНЦИПОВ КЕРКГОФФСА

- ***Физическая и математическая стойкость:*** *«Система должна быть физически, если не математически, невскрываемой»*
- ***Открытость алгоритма:*** *«Нужно, чтобы не требовалось сохранение системы в тайне; попадание системы в руки врага не должно причинять неудобств»*
- ***Возможность выбора и обмена ключами:*** *«Хранение и передача ключа должны быть осуществимы без помощи бумажных записей; корреспонденты должны располагать возможностью менять ключ по своему усмотрению»*
- ***Пригодность для каналов связи:*** *«Система должна быть пригодной для сообщения через телеграф»*
- ***Компактность:*** *«Система должна быть легко переносимой, работа с ней не должна требовать участия нескольких лиц одновременно»*
- ***Простота:*** *«Наконец, от системы требуется, учитывая возможные обстоятельства её применения, чтобы она была проста в использовании, не требовала значительного умственного напряжения или соблюдения большого количества правил»*

При построении прототипа и написании программ мы старались придерживаться шести принципов Керкгоффса

ОПИСАНИЕ ОДНОРАЗОВОГО БЛОКНОТА (ONE-TIME PAD или OTP)

- *Пример: алфавит из 50 символов*

"	0	Д	5	И	10	Н	15	Т	20	Ч	25	Ь	30	1	35	6	40	.	45
А	1	Е	6	Й	11	О	16	У	21	Ш	26	Э	31	2	36	7	41	?	46
Б	2	Ё	7	К	12	П	17	Ф	22	Щ	27	Ю	32	3	37	8	42	-	47
В	3	Ж	8	Л	13	Р	18	Х	23	Ъ	28	Я	33	4	38	9	43	:	48
Г	4	З	9	М	14	С	19	Ц	24	Ы	29	0	34	5	39	!	44		49

- *Пример: шифрование одноразовым блокнотом полученным от ГСЧ*

Текст	Д	О		В	С	Т	Р	Е	Ч	И		В		В	О	Р	О	Н	О	В	О	!
Код текста	5	16	49	3	19	20	18	6	25	10	49	3	49	3	16	18	16	15	16	3	16	44
Текст ГСЧ	З	Ш	А	Х	П	Р	:	0	6	Э	Т	4	8	Ь	0	Ж	Ц	2	И	Ф		М
Код от ГСЧ	9	26	1	23	17	18	48	34	40	31	20	38	42	30	34	8	24	36	10	22	49	14
Шифротекст	М	8	"	Ш	2	4	О	6	Н	7	С	7	7	Я	"	Ш	6	А	Ш	Ч	Н	Ж
Код шифротекста	14	42	0	26	36	38	16	40	15	41	19	41	41	33	0	26	40	1	26	25	15	8

Шифрование одноразовым блокнотом заключается в накладывании на открытый текст случайного ключа той же или большей длины

НЕМНОГО ИСТОРИИ

1882	***Изобретатель** – американский банкир **Франк Миллер (1882)***
1919	*US Patent 1,310,719 «Секретная сигнальная система» **Гильберта Вернама** («шифр Вернам»)*
1922	*Джозеф Мабурн (США) предположил, что **если пользоваться абсолютно случайным ключом и использовать его только один раз, то криптоанализ системы не даст результатов.***
1923	***Германский МИД переходит на использование одноразовых блокнотов***
1927	***СССР переходит на использование одноразовых блокнотов***
1930-е	***Военные, разведка и спецслужбы по всему миру широко используют одноразовые блокноты***
1941	*Владимир Александрович Котельников (СССР) доказал, что **системы шифрования с одноразовыми ключами (или одноразовыми блокнотами) являются абсолютно стойкими***
1941	***В США запущена военная голосовая система** на основе одноразового блокнота (SIGSALY)*
1943	***В США запущен сверхсекретный проект Venona** для расшифровки перехваченных советских сообщений*
1945	***Клод Шеннон (США) пришел к тем же выводам,** что и Котельников*
1949	***Клод Шеннон (США) опубликовал работу «Теория связи в секретных системах»***
1950-е	***«Номерные радиостанции» начинают вещать по всему миру***
1970	*Стивен Визнер (США) предлагает идею квантовой криптографии*
1982	*Доклад Чарльза Беннета (Канада) с др. «Квантовая криптография II: **как безопасно снова использовать одноразовый блокнот** даже если P=NP»*
1984	*Опубликован BB84 – первый протокол квантовой криптографии и квантового распределения ключей, тесно связанный с использованием одноразовых блокнотов*
2015	*АНБ США **объявило о переходе к пост-квантовой криптографии** в течение 10 лет*

В 1941 году Котельников и в 1945 году Шеннон доказали абсолютную криптостойкость одноразовых блокнотов

АБСОЛЮТНО СТОЙКИЙ ШИФР

- *Котельников в 1941 и Шеннон в 1945 году доказали, что шифр является абсолютно стойким, если соблюдаются следующие условия:*

  (1) ***Секретность:*** *«Ключ секретен — известен только легитимным пользователям»*

  (2) ***Превышение длины ключа над длиной сообщения:*** *«Длина ключа в битах не меньше длины сообщения»*

  (3) ***Случайность:*** *«Ключ случаен»*

  (4) ***Одноразовость:*** *«Ключ используется только один раз»*

Котельников и Шеннон сформулировали требования к абсолютно стойкому шифру, которым удовлетворяет только одноразовый блокнот

ПРЕИМУЩЕСТВА ОДНОРАЗОВЫХ БЛОКНОТОВ

- ***Абсолютная криптостойкость*** *– невозможность расшифровки сообщения после его перехвата противником никакими методами, даже если противник обладает неограниченным вычислительным ресурсом*
- ***Степенной рост трудности*** *порядка A^M для лобовой «атаки грубой силой» при росте длины сообщения, в то время как у традиционных схем шифрования трудность постоянна и равна A^P (где A – длина алфавита, P – длина пароля, M – длина сообщения)*
- ***Абсолютная энтропия*** *– равная вероятность любого возможного варианта открытого текста той же длины*
- ***Невосстанавливаемость сообщений*** *– даже физический доступ противника к одноразовому блокноту, у которого уничтожены предыдущие «страницы», не позволяет восстановить прошлые сообщения (так называемая «совершенная секретность в будущем»)*
- ***Возможность использования в системах супершифрования*** *в качестве одного из слоев шифрования или системы шифрования длинного пароля*
- ***Открытая передача сообщений*** *– сообщения можно свободно передавать по любым открытым сетям*
- ***Полная возможность «правдоподобного отрицания»*** *в случае компрометации пользователя*
- ***Быстрота шифрования*** *– шифрование с помощью одноразового блокнота требует значительно меньше операций, чем при помощи традиционных шифров (например, DES, AES или RSA)*

Одноразовые блокноты имеют целый ряд важных преимуществ, включая абсолютную криптостойкость и энтропию сообщений

НЕДОСТАТКИ ОДНОРАЗОВЫХ БЛОКНОТОВ

- ***Требование полной случайности*** *– труднодостижимое на практике, так как нет хороших генераторов случайных чисел*
- ***Требование абсолютной одноразовости*** *– невозможности полного уничтожения всех копий блокнота после использования (касается остаточной информации на носителях данных)*
- ***Требования безопасного распределения ключей*** *– невозможно быстрое осуществление в необходимых объемах (требуется физическая доставка или использование других менее криптостойких методов шифрования блокнотов для доставки по сетям)*
- ***Требование аутентификации отправителя*** *– необходимость дополнительной и менее стойкой системы для аутентификации отправителя*
- ***Большая длина –*** *традиционная криптография сокращает размер секрета до размера ключа, в то время как одноразовый блокнот требует их равенства*
- ***Непрактичность для аудио и видео –*** *традиционная криптография успешно шифрует большие куски информации (аудио, видео, VPN и Интернет-соединения) и использование одноразовых блокнотов непрактично для этих целей*
- ***Проблема распределения ключей –*** *одноразовый блокнот не решает основную проблему современной криптографии – безопасное распределение ключей, а только усложняет ее*
- ***Ненужная более высокая криптостойкость –*** *одноразовый блокнот решает проблему повышения криптостойкости шифров, которая и так достаточна высока.*

При традиционном взгляде на шифрование одноразовые блокноты имеют целый ряд недостатков

КВАНТОВАЯ КРИПТОГРАФИЯ И БУДУЩЕЕ ОДНОРАЗОВЫХ БЛОКНОТОВ

- ***Сумасшедшая идея*** *использовать квантовые состояния фотона для маркировки денег (1970-е годы, «Сопряженное кодирование», Стевен Визнер)*
- *«Квантовая криптография II:* ***как безопасно использовать снова одноразовый блокнот*** *даже если P=NP» (1982 год, Жиль Брассар и Сет Брейдбарт)*
- ***Квантовое распределение ключей*** *(протокол BB84, 1984, Жиль Брассар и Сет Брейдбарт)*
- *Квантовая криптография* ***основана на законах квантовой механики****, а не на трудностях решения математических проблем*
- *При использовании канала передачи, основанном на квантовом состоянии фотонов,* ***перехват данных и прослушивание становится невозможным, поскольку неизбежно ведет к обнаружению и безвозвратному изменению*** *перехваченных данных.*
- *Верифицированный* ***одноразовый блокнот*** *(или ключ), полученный по такому каналу,* ***может использоваться снова и снова*** *для шифрования сообщений пока не выяснится, что он перехвачен противником.*
- *При обнаружении перехвата, блокнот (или ключ)* ***заменяется на другой, до тех пор, пока не обеспечивается и подтверждается его верифицированное получение.***
- *Квантовая криптография тесно связана с идеей использования* ***одноразовых блокнотов, как наиболее естественного способа существования ключей и шифрования сообщений.***
- ***Квантовая криптография быстро развивается*** *и начинает влиять на традиционные методы шифрования*

Квантовая криптография тесно связана с шифрованием одноразовыми блокнотами и решает раз и навсегда проблему их безопасного распространения

СУПЕРШИФРОВАНИЕ

- *Эффективное удлинение ключей при каскадном шифровании*
- *Последовательное использование нескольких алгоритмов (например, 16 раз AES или AES-Serpent-Twofish)*
- *Достаточно хорошо изучено при переходе от DES к 3DES*
- *В 2014 году международной группой математиков было показано, что при каскадном шифровании эффективное удлинение ключа определяется формулой:*

$$2^{\left(k+min\left\{k*\left(\frac{L-2}{2}\right);\frac{n(L-2)}{L}\right\}\right)} \Rightarrow max \equiv 2^{(k+n)}, L \to \infty$$

K –длина ключа, n – длина блока, L – количество раундов шифрования

Прирост криптостойкости и эффективной длины ключа с увеличением раундов шифрования для симметричного блокового шифра с размером блока n=128

n=128	Количество раундов шифрования									
	2	4	16	32	64	128	256	512	1024	2048
64	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
128	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
192	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
256	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
320	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
384	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
448	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9
512	0	64.0	112.0	120.0	124.0	126.0	127.0	127.5	127.8	127.9

Применение каскадного шифрования позволяет эффективно удлинить ключ и увеличить его криптостойкость

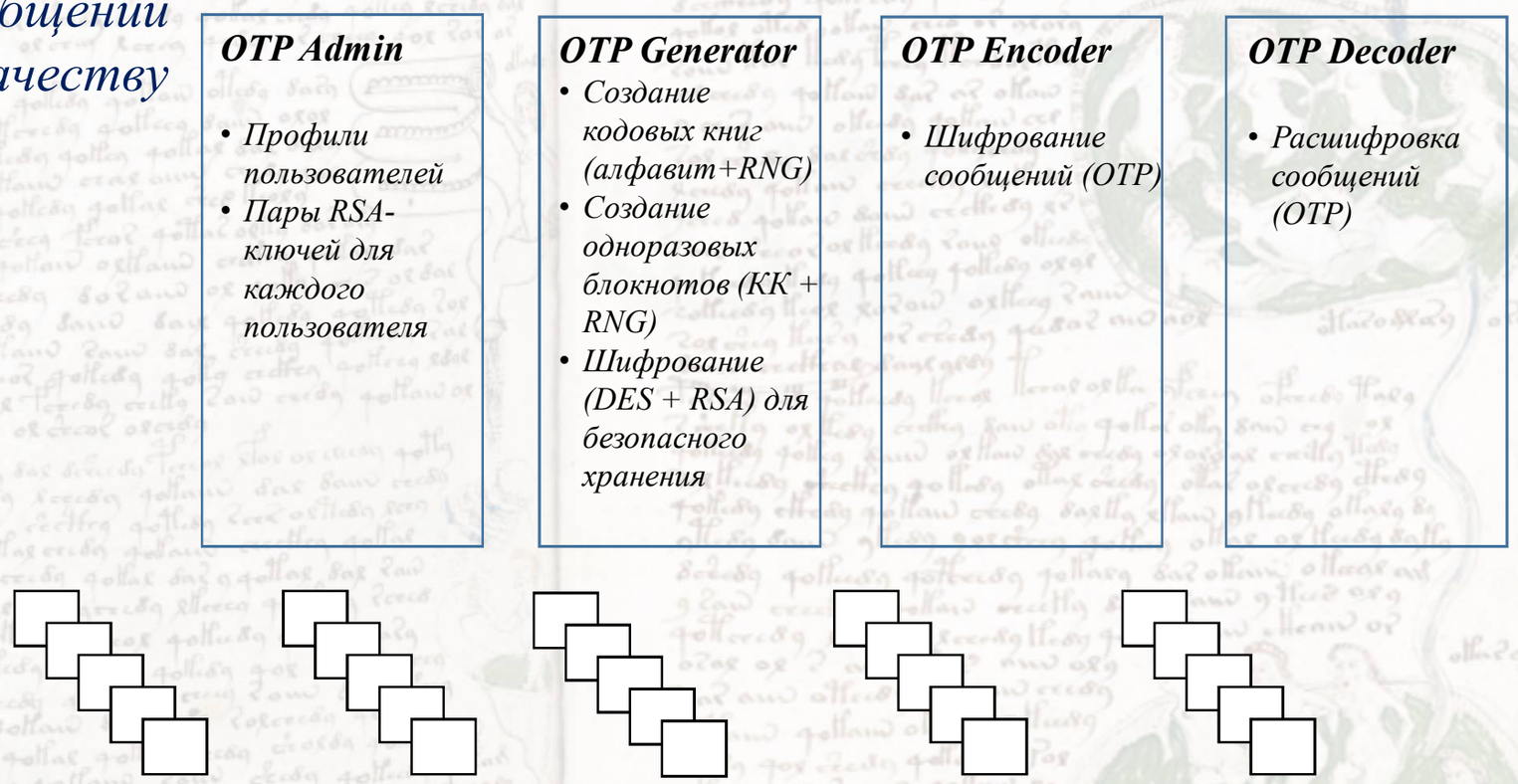
ТЕХНОЛОГИИ И АЛГОРИТМЫ

- ***Платформа:*** *Microsoft .NET Framework 4.7*
- ***Язык программирования:*** *C# 7.0*
- ***IDE:*** *Microsoft Visual Studio Professional 2017*
- ***Модули:***
  - *Стандартные модули C# 7.0*
  - *System.Security.Cryptography*
  - *System.XML*
- ***Внешние модули:***
  - *Ionic.ZIP (шифрованные ZIP-контейнеры)*
- ***ГПСЧ:***
  - *System.Random (встроенный)*
  - *System.Random Thread-Safe (встроенный с использованием ThreadLocal<T>)*
  - *RNGCryptoServiceProvider (встроенный, удовлетворяет требованиям FIPS к ГСЧ)*
  - *PCG (внешний модуль на C# и отдельная реализация на C++, близок к требованиям FIPS к ГСЧ)*
  - *MT19937 (Mersenne Twister, внешний модуль на C# и отдельная реализация на Python)*
- ***Хранение ключей RSA:*** *встроенный в Windows криптоконтейнер MachineKeyStore*
- ***Кодовые таблицы:*** *собственный алфавит из 256 символов*
- ***Физическая стойкость:*** *TrueCrypt 7.1a*

Построенный прототип опирался на набор из целого ряда технологий и алгоритмов

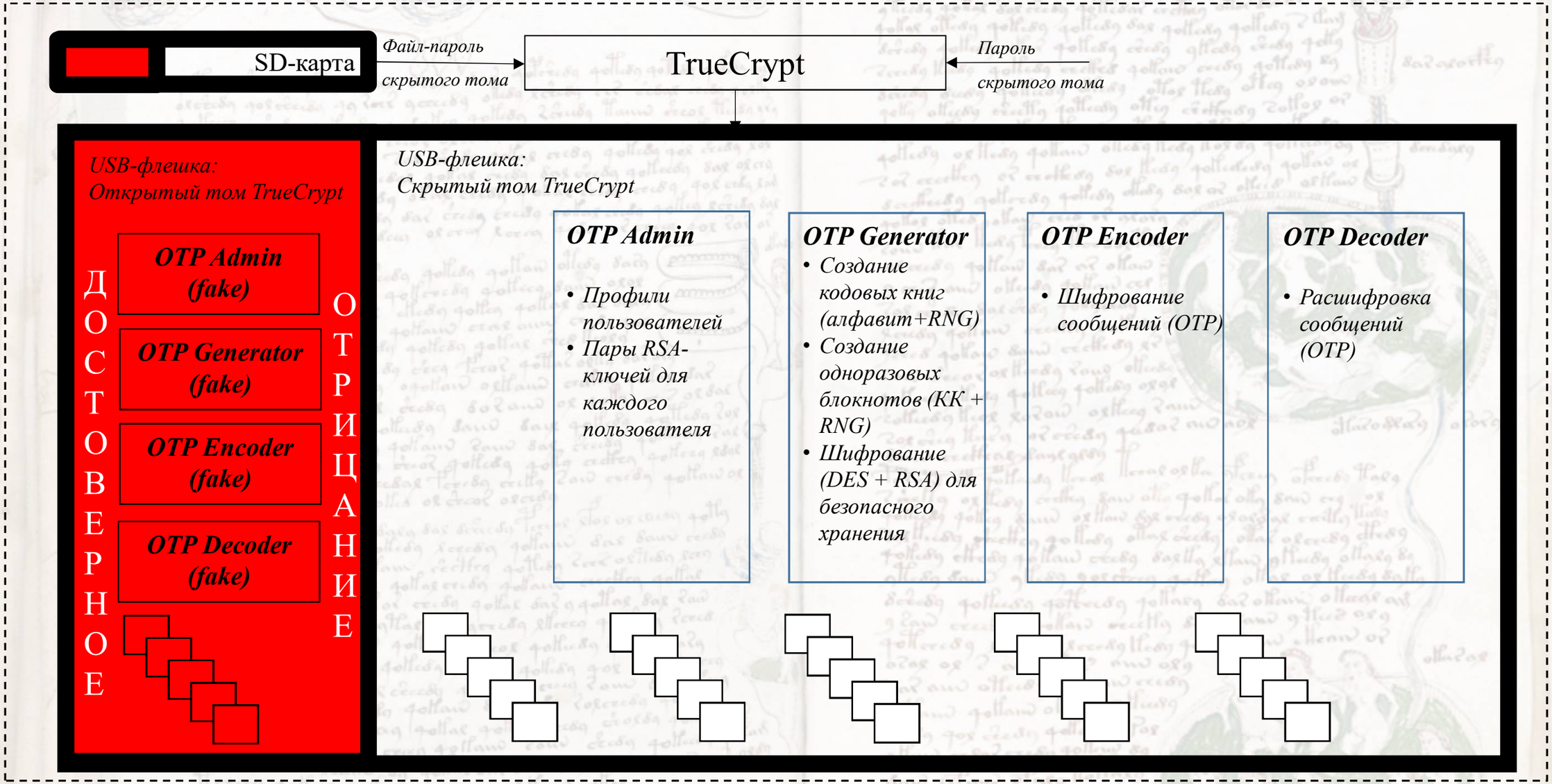
РЕАЛИЗАЦИЯ ПРОТОТИПА

- *Функциональность:*
  - *Создание профилей администраторов и пользователей*
  - *Генерация случайных кодовых таблиц и одноразовых блокнотов (пять ГСЧ)*
  - *Шифрование сообщений*
  - *Расшифровка сообщений*
- *ГСЧ отличались по качеству и скорости работы*



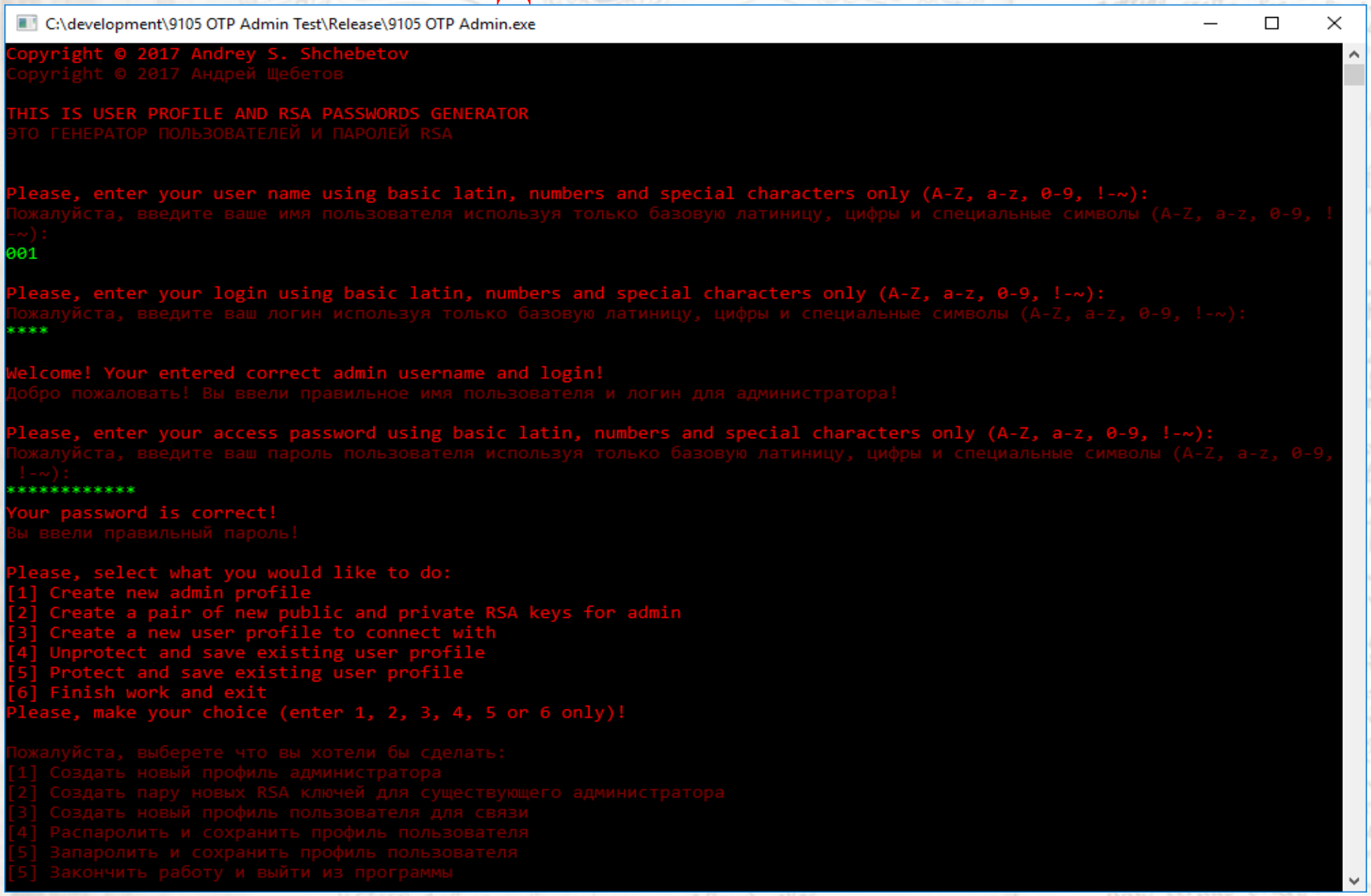
Прототип был написан на C# в виде 4 отдельных программ-модулей, каждая из которых исполняла свою выделенную задачу

РЕАЛИЗАЦИЯ ПРОТОТИПА: ФИЗИЧЕСКАЯ БЕЗОПАСНОСТЬ



Все программы расположены на скрытом томе TrueCrypt, который не может быть обнаружен никакими существующими методами

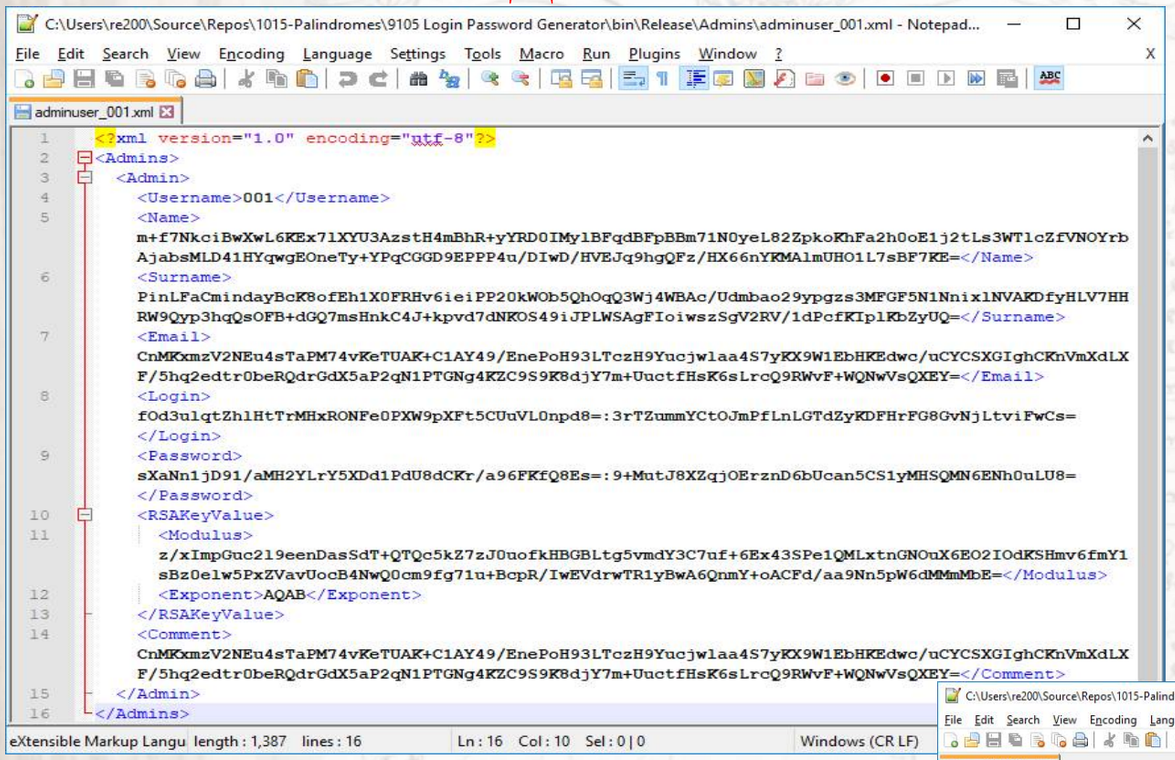
OTP ADMIN: СОЗДАНИЕ ПРОФИЛЕЙ ПОЛЬЗОВАТЕЛЕЙ



- *Создание профилей администраторов и пользователей*
- *Создание ключей RSA для администраторов*
- *Хранение ключей RSA в криптоконтейнерах Machine Key Store на компьютере*
- *Создание профилей пользователей*
- *Хранение всех профилей только в виде XML-файлов с захэшированной или запароленной информацией*

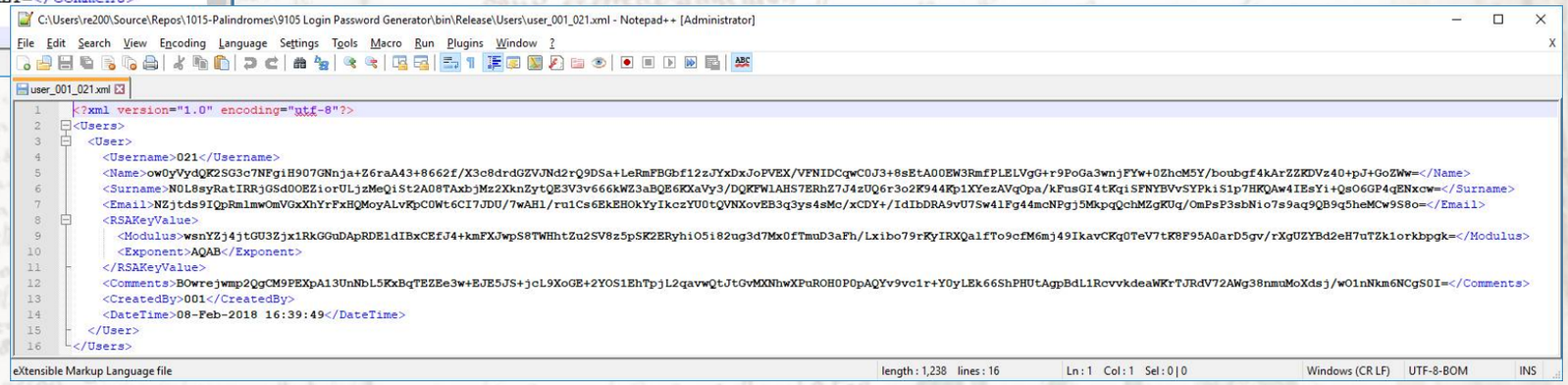
OTP Admin позволял создавать профили администраторов и пользователей системы, а также их RSA ключи

ПРОФИЛИ АДМИНИСТРАТОРА И ПОЛЬЗОВАТЕЛЕЙ



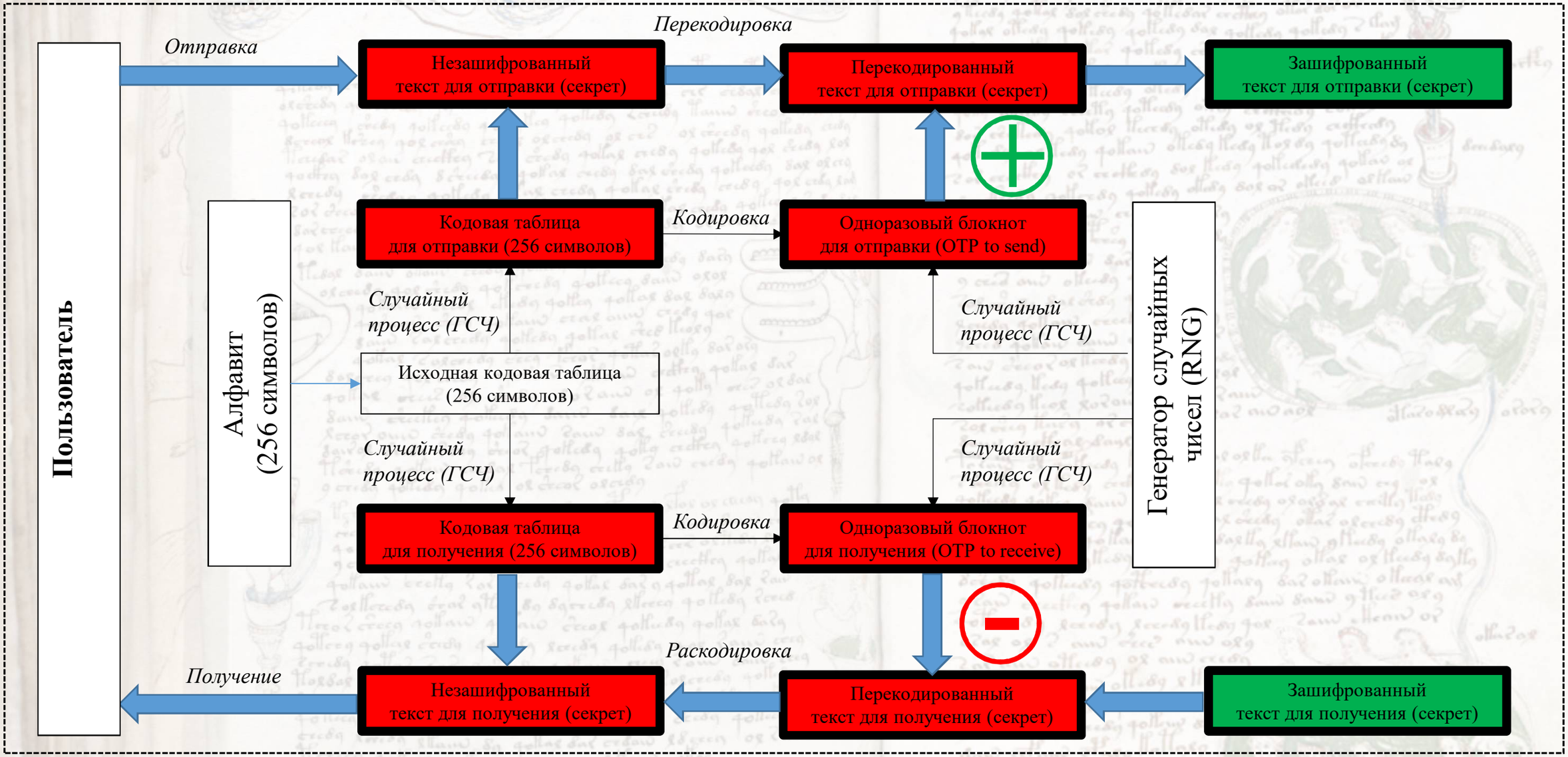
- *Вход в систему и все ее программы без существующего профиля администратора невозможен*
- *Все данные либо хэшируются либо запаролены*
- *При каждом входе вся введенная информация сравнивается с хранящейся в XML-профиле*
- *Каждый администратор, вошедший в систему, может создавать новые профили администраторов и пользователей*
- *Можно изменять пары RSA-паролей, хранящиеся в специальных криптоконтейнерах, встроенных в Windows*

- *Каждый администратор может создавать профили своих пользователей, чтобы обмениваться с ними сообщениями*



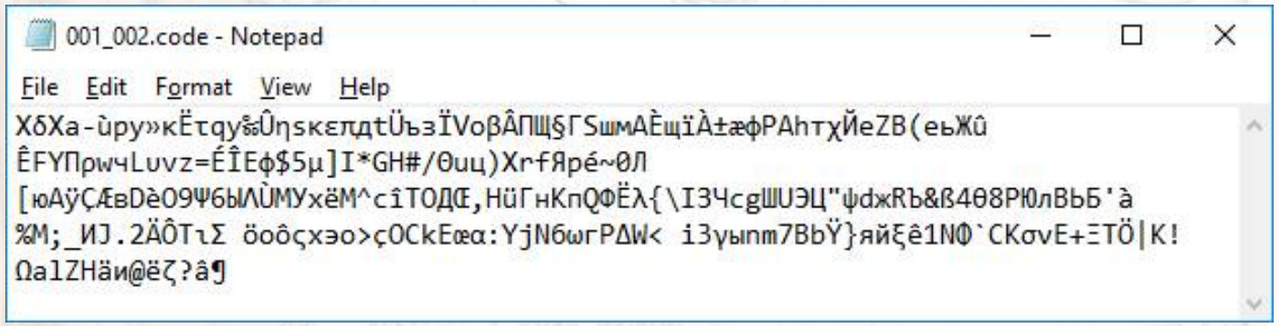
Вся информация об администраторах и пользователях хранится в XML-файлах с захэшированной или запароленной информацией

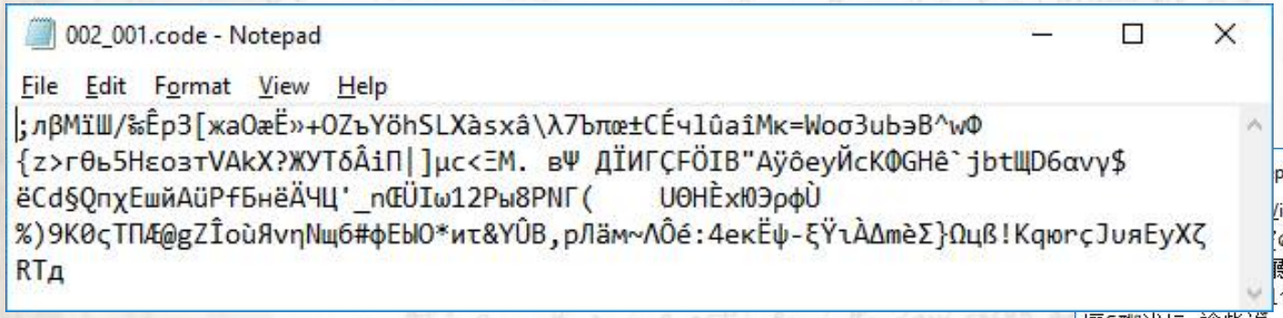
ОБЩАЯ СХЕМА ШИФРОВАНИЯ ОДНОРАЗОВЫМ БЛОКНОТОМ



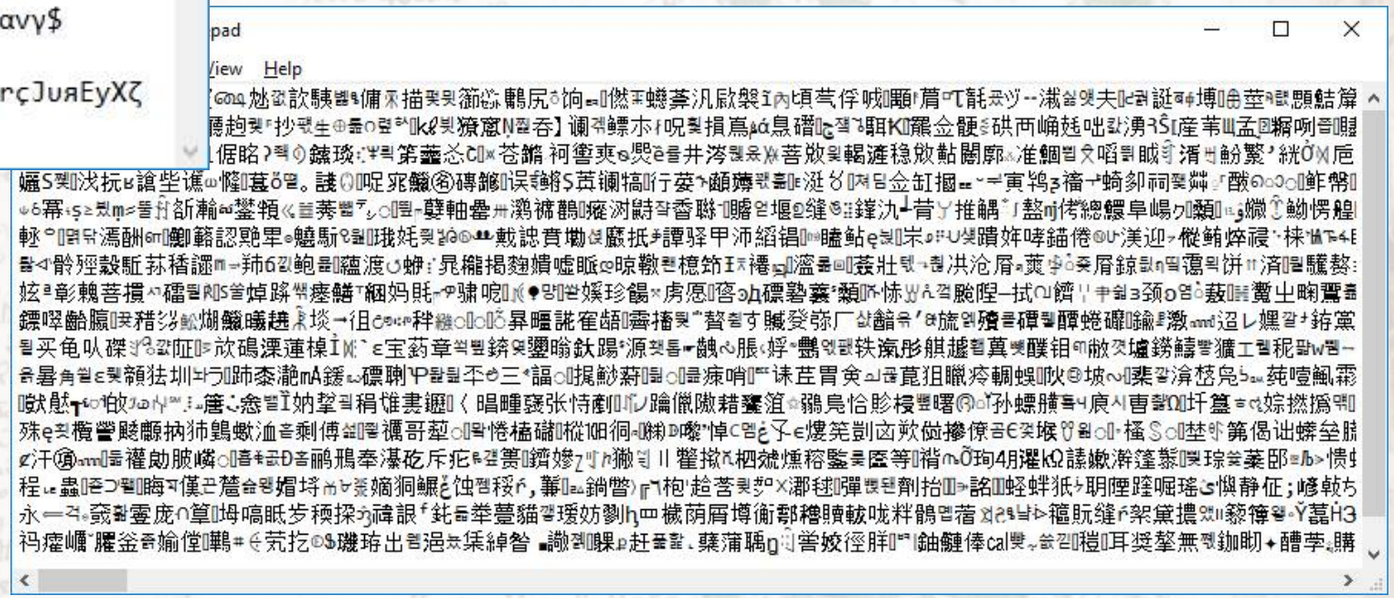
Общая схема шифрования одноразовым блокнотом зависит от нескольких случайных процессов, генерируемых набором ГСЧ

КОДОВЫЕ ТАБЛИЦЫ





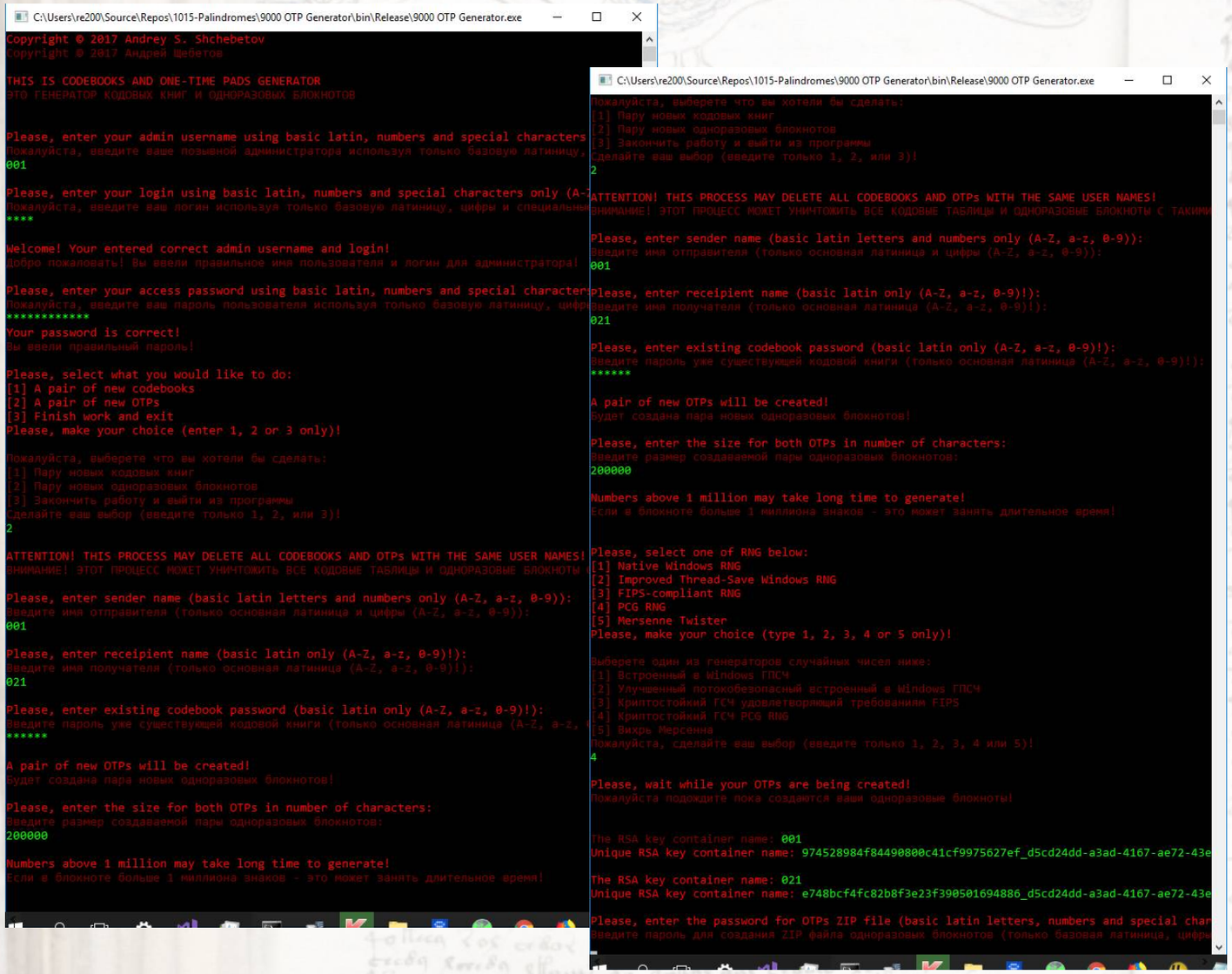
- *Примерный вид пары кодовых таблиц на основе собственного алфавита из 256 символов*
- *256 совпадает с цифровым значением байтов (0-255)*
- *Позволяет писать на русском, английском, немецком, французском и греческом языках*
- *Символы полностью перепутаны, нет никакого порядка*
- *Смена кодовой таблицы эквивалентна смене шифра*



- *Вид зашифрованного текста при использовании алфавита из 55295 одиночных («несуррогатных») символов UTF-16*
- *Большой алфавит труднее расшифровать, но требуется больше памяти*
- *Проверялись также сокращенные варианты (например, Base64 или алфавит из 50 символов)*

Кодовые таблицы были основаны на собственном алфавите из 256 символов, который может быть расширен до 55295 одиночных символов UTF-16

OTP GENERATOR: ГЕНЕРАЦИЯ ТАБЛИЦ И ОДНОРАЗОВЫХ БЛОКНОТОВ



- *Генерация кодовых таблиц (для каждой пары пользователей одна «на отправку» и одна «на получение»)*
- *Генерация одноразовых блокнотов (для каждой пары пользователей один «на отправку» и один «на получение»)*
- *Пять встроенных ГПСЧ, один из которых удовлетворяет требованиям FIPS и один (PCG) – близок к ним*
- *ГСЧ отличались качеством и скоростью работы*
- *Сохранение кодовых таблиц и одноразовых блокнотов в запароленных криптоконтейнерах на скрытом томе TrueCrypt*

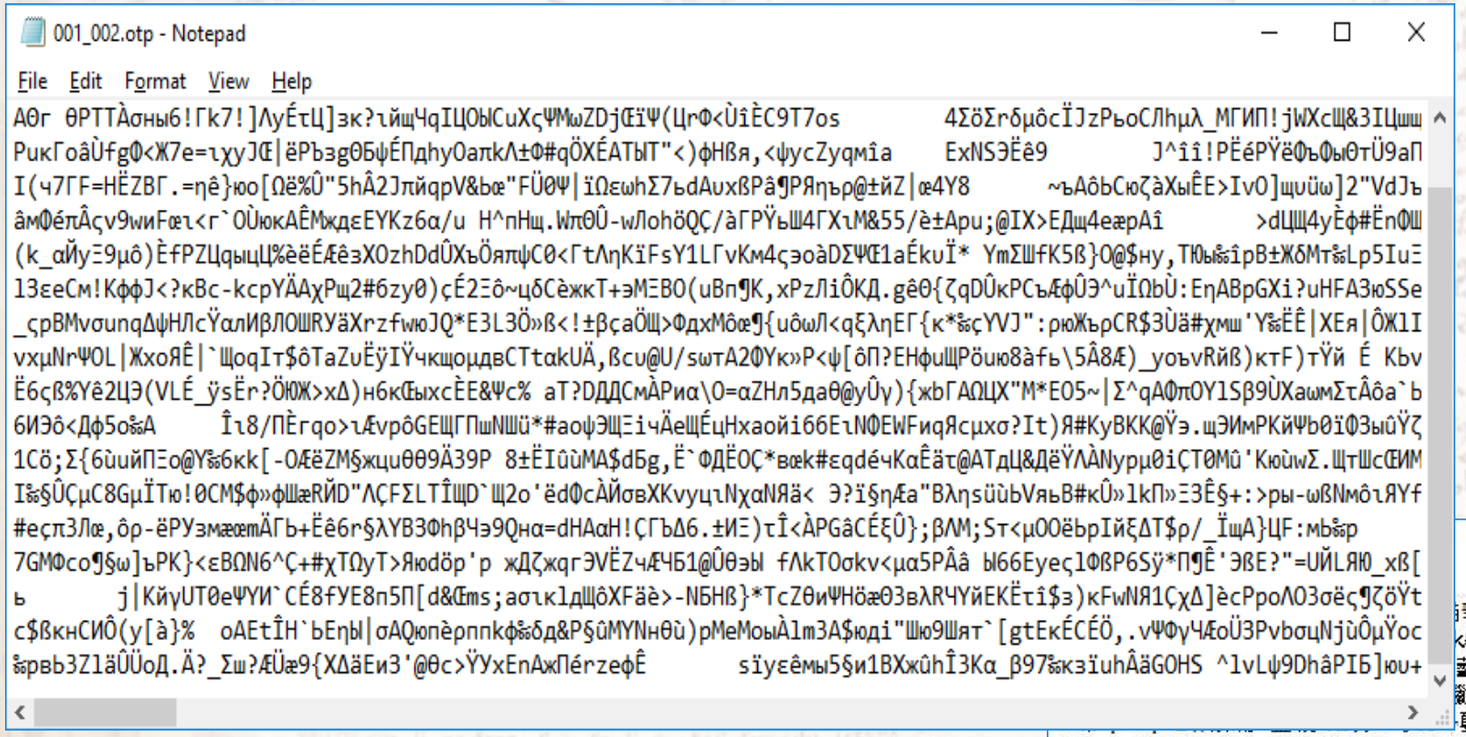
OTP Generator позволял генерировать кодовые таблицы и одноразовые блокноты с использованием 5 генераторов псевдослучайных чисел

ОДНОРАЗОВЫЙ БЛОКНОТ ПОСЛЕ КОДИРОВКИ

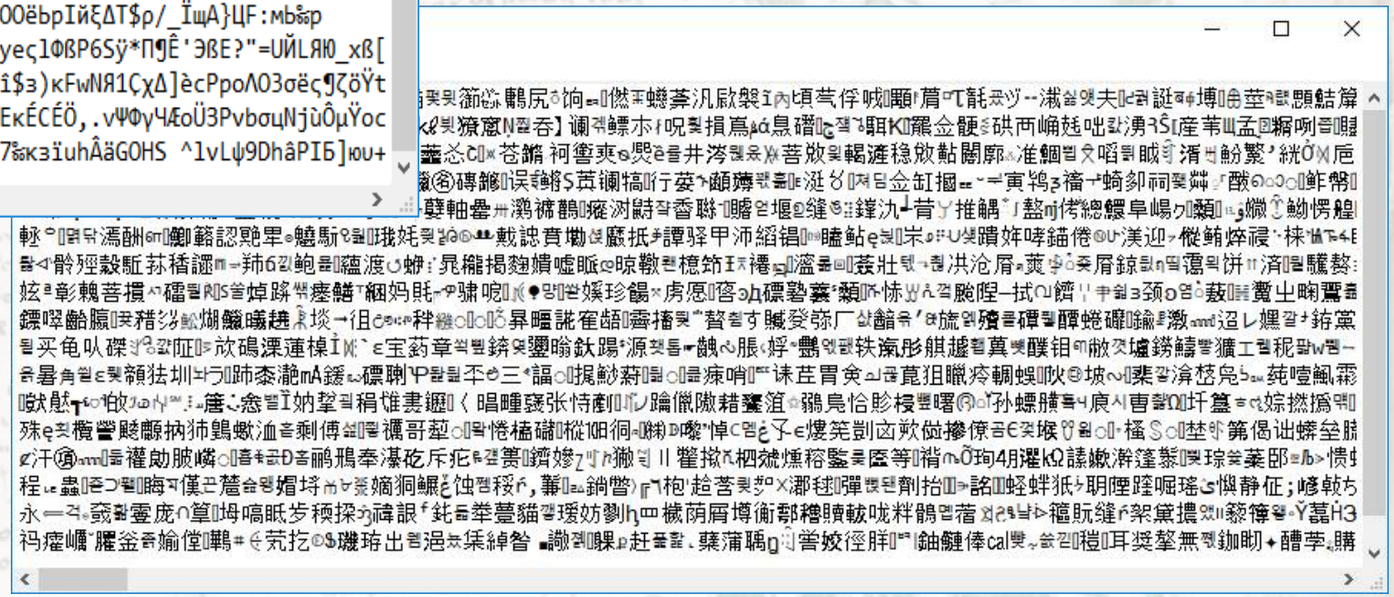
- *Сплошной случайный набор символов без каких-либо привязанных цифровых значений*
- *Возможность замены на новый блокнот в любой момент*
- *Возможность «переиспользования» блокнота при замене кодовой таблицы*
- *Хранение и пересылка только в запароленном AES-контейнере*
- *Использование – только вместе с кодовой таблицей – никогда в одиночку*

Одноразовый блокнот после кодировки выглядит как случайный текст без слов и предложений из 256 повторяющихся символов

ОДНОРАЗОВЫЙ БЛОКНОТ ПОСЛЕ СУПЕРШИФРОВАНИЯ



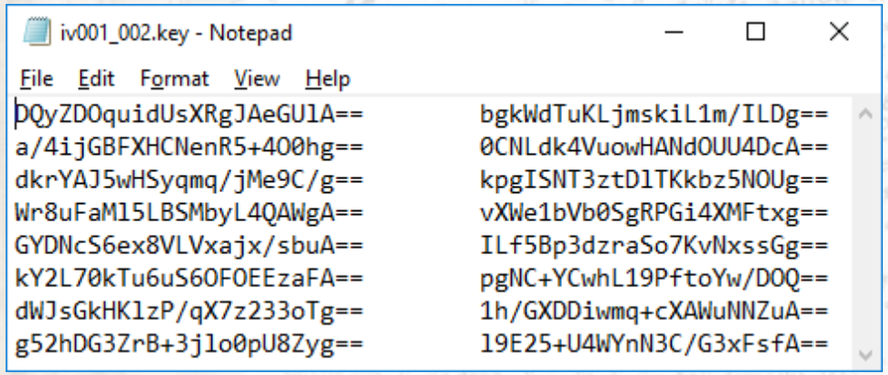
- *Одноразовый блокнот был полностью пригоден для супершифрования*
- *Вид одноразового блокнота после 16 раундов шифрования AES и «обертывания» в кодовую таблицу*
- *Тот же беспорядочный и бессистемный набор из 256 символов*

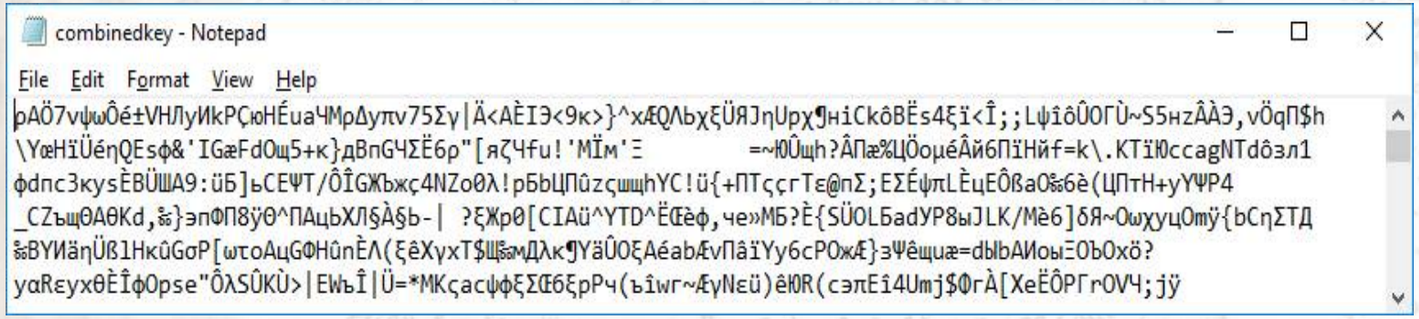


- *Вид одноразового блокнота при использовании 55295 символов кодировки UTF-16*

Сам одноразовый блокнот может быть зашифрован с помощью супершифрования с использованием традиционных алгоритмов

ВИД УДЛИНЕННОГО КЛЮЧА ПОСЛЕ СУПЕРШИФРОВАНИЯ

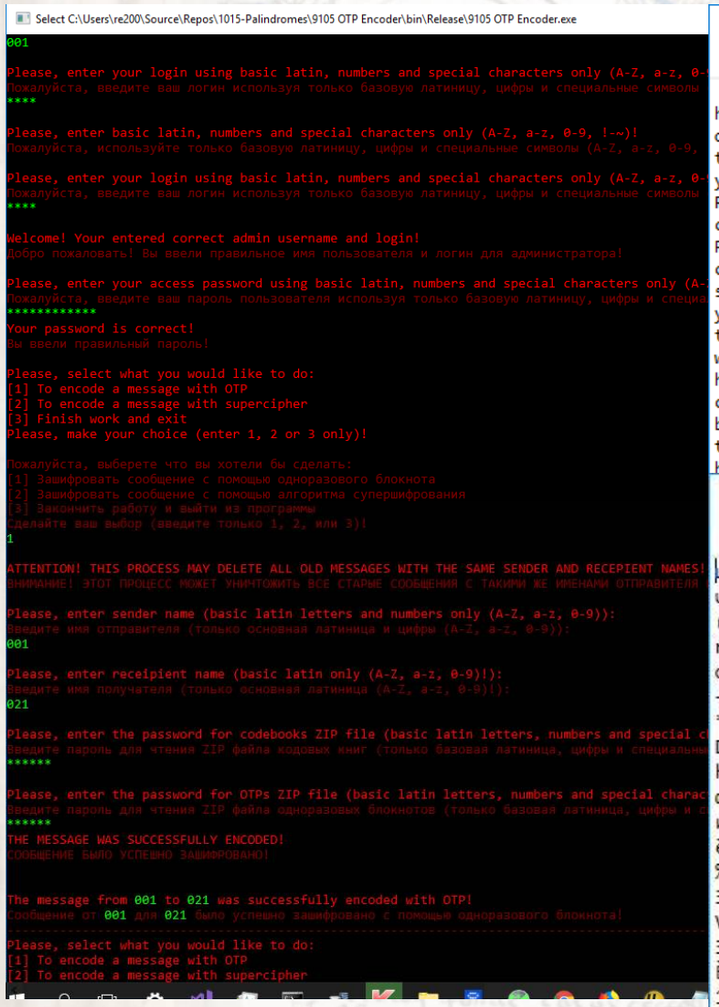


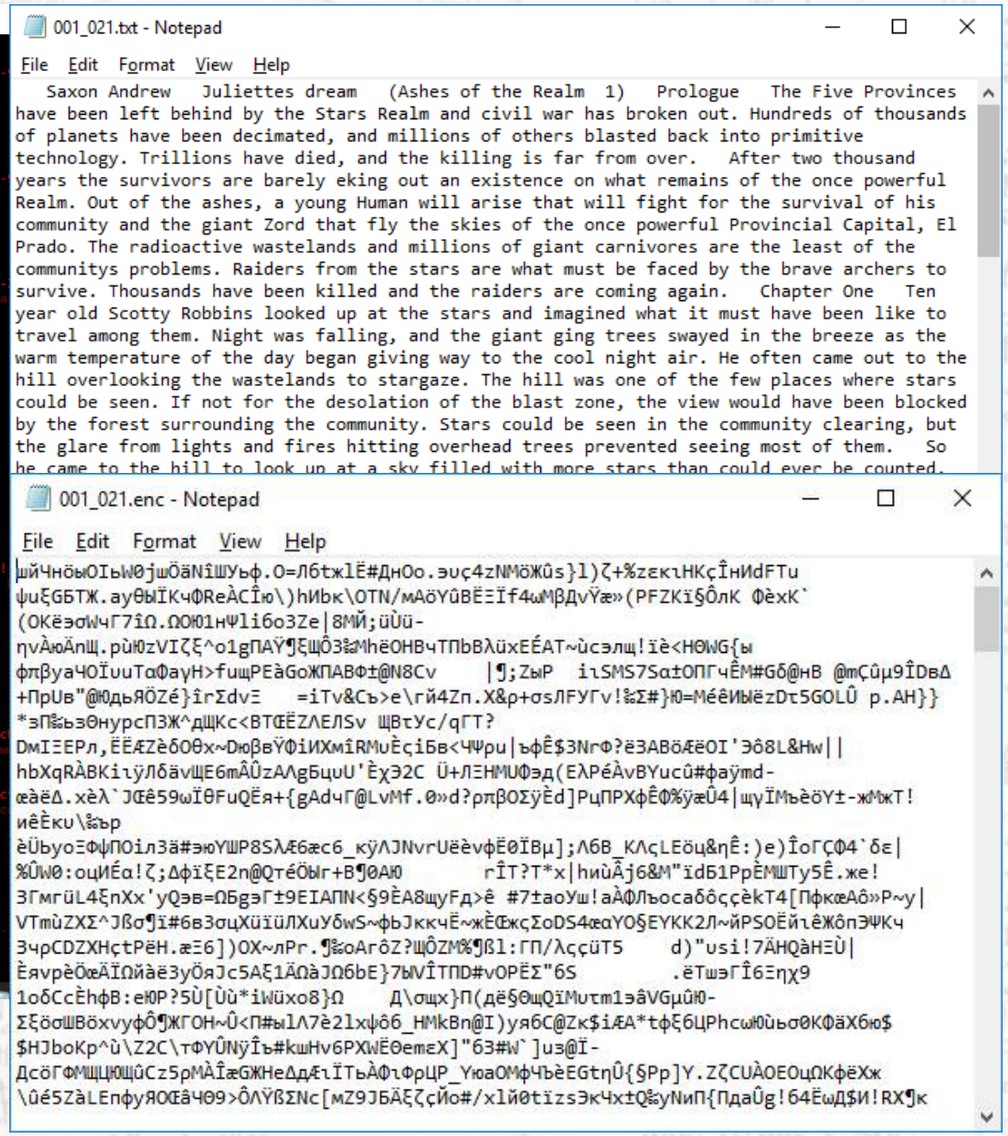


- *Примерный ключа в кодировке Base64 до шифрования оригинального ключа кодовой таблицей*
- *Примерный вид векторов инициализации в кодировке Base64 до шифрования оригинального ключа кодовой таблицей*
- *Типичный вид ключа после применения кодовой таблицы и одноразового блокнота*

Удлиненный ключ после супершифрования выглядел как текст из 6144 символов кодировочной таблицы

OTP ENCODER: ШИФРОВАНИЕ СООБЩЕНИЙ

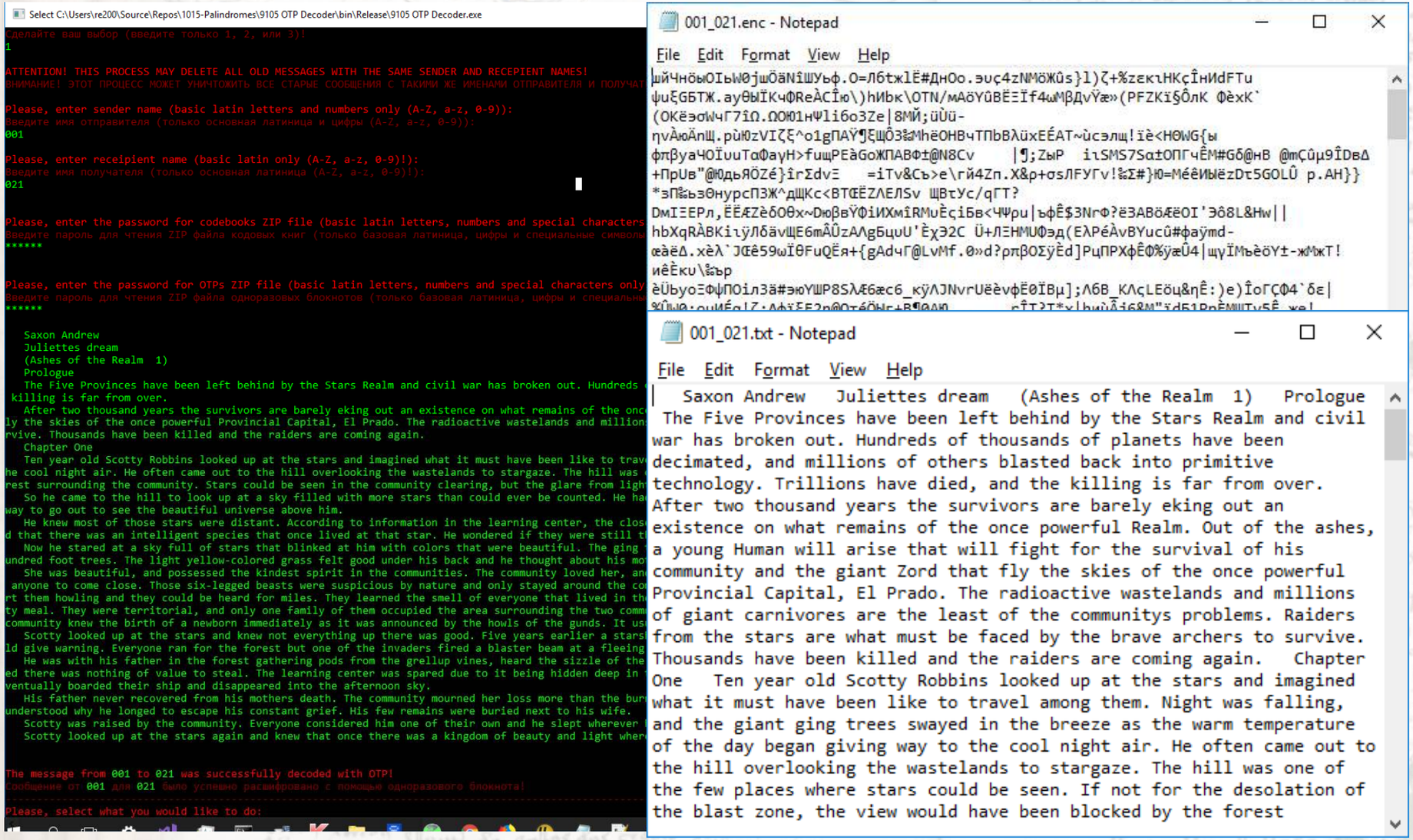




- *Пример шифрования с помощью одноразового блокнота*
- *На входе – англоязычный текст*
- *На выходе невозможно определить язык, разбить на слова и предложения, понять, где знаки препинания*
- *При шифровании одноразовым блокнотом одного и того же текста на выходе всегда будет другой шифротекст*

OTP Encoder позволял шифровать открытый текст с помощью одноразового блокнота в шифротекст из которого невозможно было определить даже язык

OTP DECODER: РАСШИФРОВКА СООБЩЕНИЙ

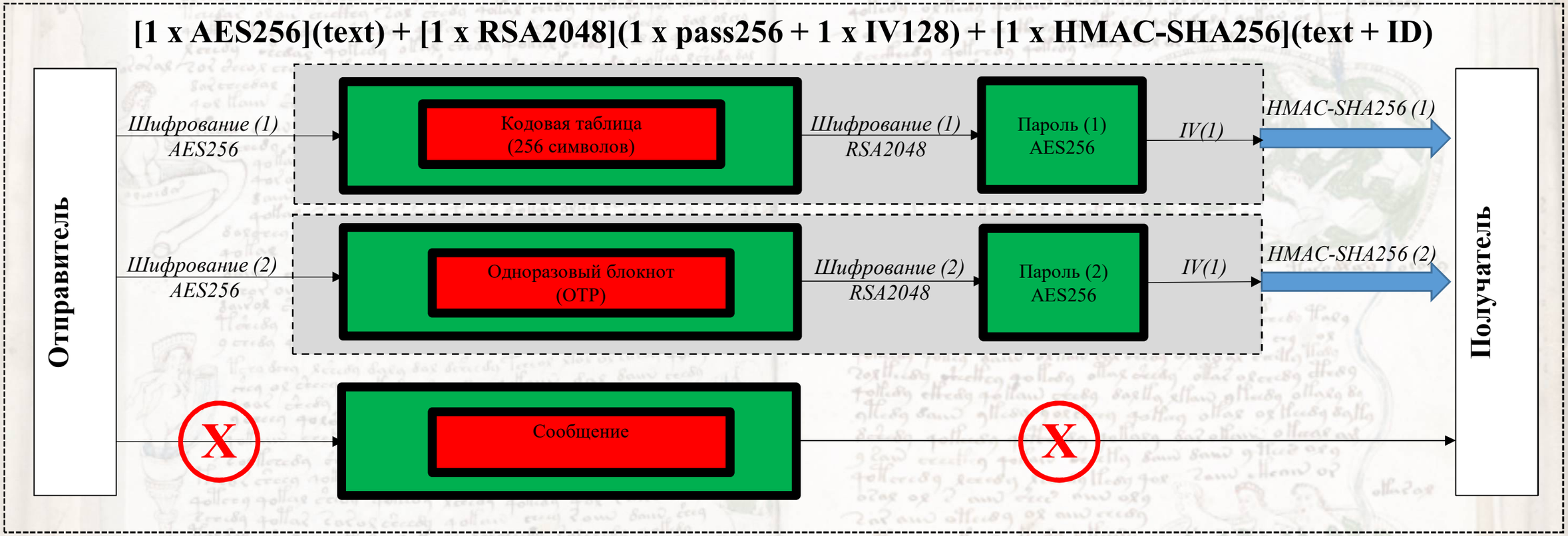


- *Пример зашифрованного сообщения*
- *Расшифровка выполнялась в обратном порядке*
- *Сообщение записывалось в виде текстового файла и выводилось на экран, а также записывалось в соответствующую директорию на хард диске*

OTP Decoder позволял расшифровывать сообщения, зашифрованные с помощью одноразового блокнота

ПРОТОКОЛЫ: ОБМЕН ПРИ ОТСУТСТВИИ ОБЩИХ ТАБЛИЦ И БЛОКНОТОВ

ОБМЕН КОДОВЫМИ ТАБЛИЦАМИ, ОДНОРАЗОВЫМИ БЛОКНОТАМИ И СООБЩЕНИЯМИ: ВАРИАНТ 1

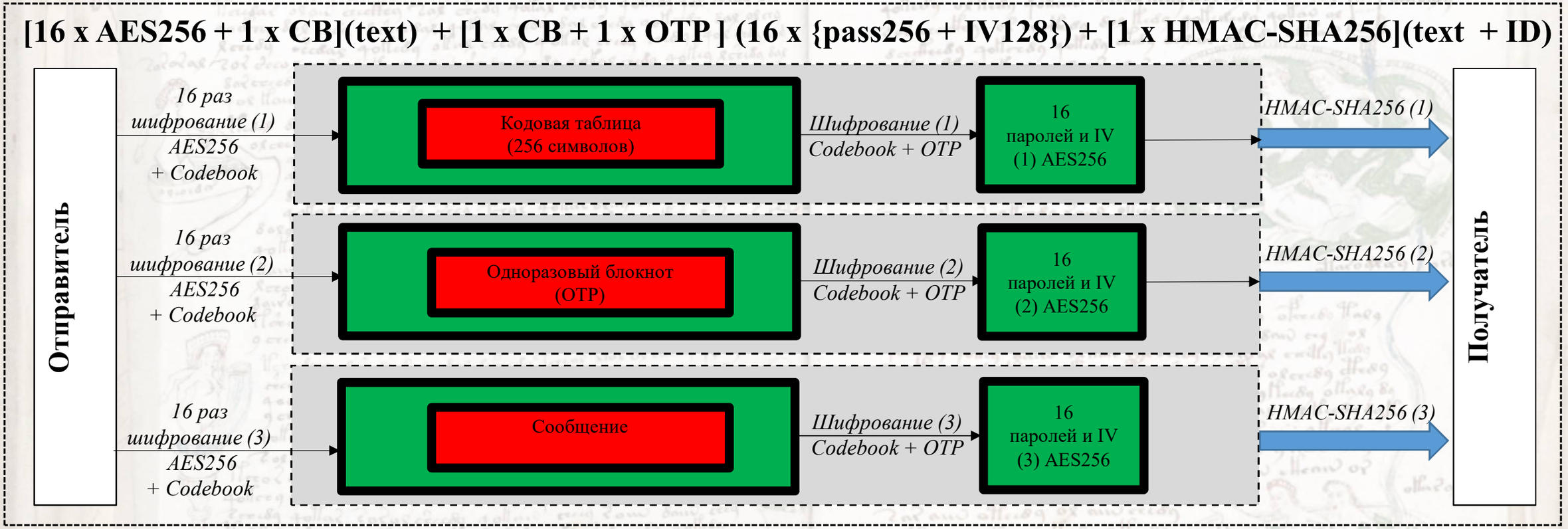


При отсутствии у пользователей общих кодовых таблиц и одноразовых блокнотов они могут ими обменяться традиционным способом: AES + RSA

ПРОТОКОЛЫ: ОБМЕН ПРИ СУПЕРШИФРОВАНИИ

ОБМЕН КОДОВЫМИ ТАБЛИЦАМИ, ОДНОРАЗОВЫМИ БЛОКНОТАМИ И СООБЩЕНИЯМИ: ВАРИАНТ 2

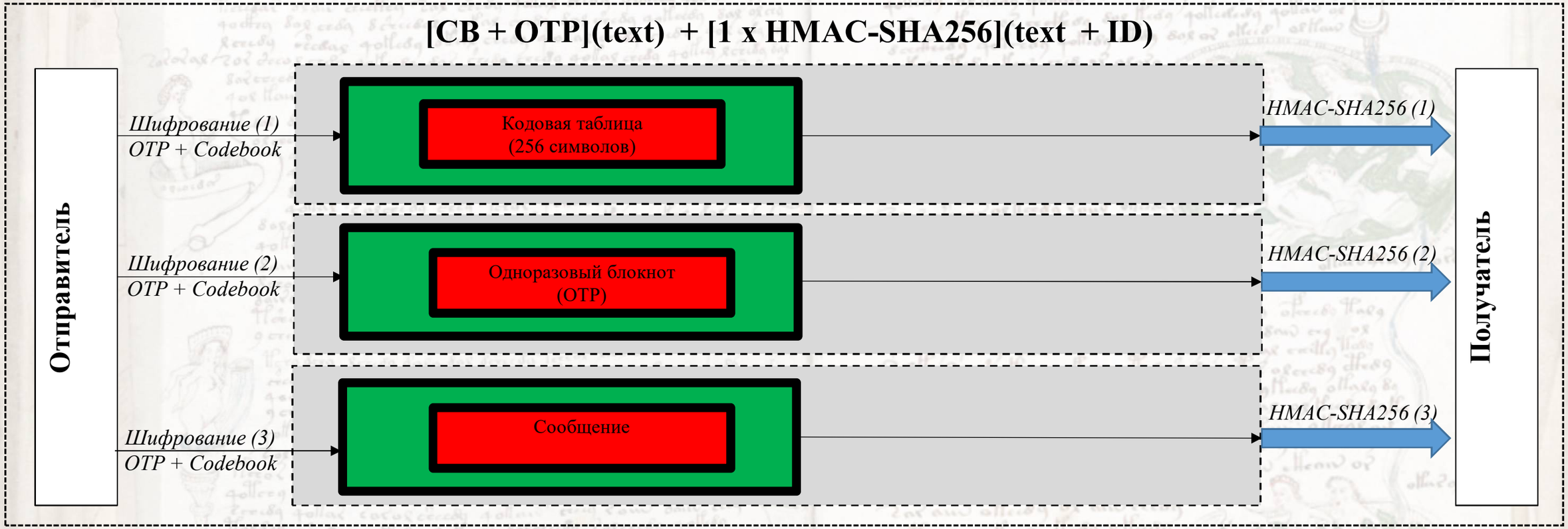
[16 x AES256 + 1 x CB](text) + [1 x CB + 1 x OTP] (16 x {pass256 + IV128}) + [1 x HMAC-SHA256](text + ID)



При наличие общих кодовых таблиц и небольших одноразовых блокнотов обмен может идти по протоколу супершифрования для экономии блокнота

ПРОТОКОЛЫ: ОБМЕН С ИСПОЛЬЗОВАНИЕМ ОДНОРАЗОВЫХ БЛОКНОТОВ

ОБМЕН КОДОВЫМИ ТАБЛИЦАМИ, ОДНОРАЗОВЫМИ БЛОКНОТАМИ И СООБЩЕНИЯМИ: ВАРИАНТ 3



При наличие достаточно большого одноразового блокнота весь обмен идет только с его помощью, что является самым простым и эффективным способом связи

РЕЗУЛЬТАТЫ

- ***Построен прототип системы шифрования*** *на основе одноразовых блокнотов, кодовых таблиц и супершифрования (включая 4 написанные на C# программы-модуля)*
- ***Создан собственный алфавит*** *из 256 символов для создания кодовых таблиц*
- ***Рассмотрены другие*** *возможные* ***алфавиты***
- ***Предложено три протокола*** *обмена кодовыми таблицами и одноразовыми блокнотами*
- ***Проверены 5 генераторов случайных чисел***
- ***Проверена работоспособность системы*** *в различных вариантах и сочетаниях*
- ***Проверена схема супершифрования*** *с использованием AES, одноразовых блокнотов и кодовых таблиц*
- ***Обеспечена физическая безопасность и наличие «достоверного отрицания»*** *сборки при помощи TrueCrypt*
- ***Дальнейшие шаги:***
  - *Доработка прототипа до полной функциональности*
  - *Разработка полноценного ПО с GUI на базе прототипа*
  - *Внедрения одноразовых блокнотов в систему обмена одноразовыми сообщениями типа PubNub*

В результате реализации проекта построен прототип системы шифрования на основе одноразовых блокнотов, который можно дорабатывать дальше