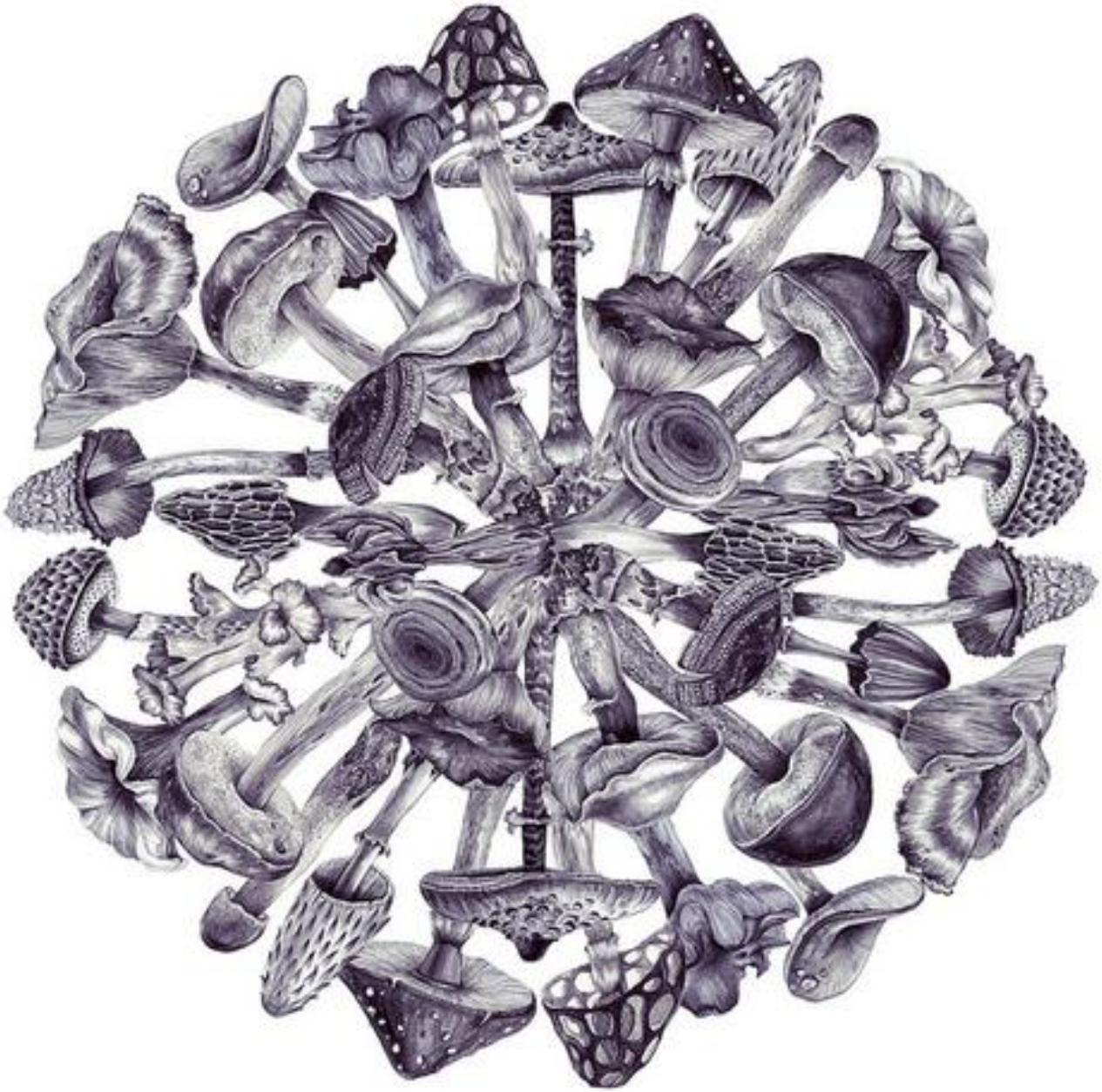

ШИФРОВАНИЕ

FY 2013

Описание алгоритма шифрования, способа реализации, вариантов использования

СОДЕРЖАНИЕ

Содержание

Модифицированный HPC (базовая шифрация)

ОПИСАНИЕ АЛГОРИТМА

Абсолютно произвольный размер шифруемого блока и ключа шифрования. Ключи шифрования также шифруются. Данный модифицируемый алгоритм использует 64кБ ключ с переменным размером шифруемого блока и одним дополнителшьным ключом – суперключом. Назначение суперключа – модифицировать результат шифрования при одинаковых значениях входных данных и ключа шифрования. Таким образом суперключ играет роль вектора инициализации в режиме шифрования со сцеплением блоков шифра. Такой «внутренний» по отношению к алгоритму вектор инициализации является большим преимуществом алгоритма и модифицируется перед шифрованием каждого блока данных путем смещения первоначального суперключа вправо на единицу. Первоначальный ключ меняется время от времени Властелином Колец. Первоначальный ключ может не задаваться вовсе. Также имеется защита самого алгоритма шифрования, бинарный код которого компилируется на основе трех отдельно введенных кодовых фраз.

В оригинальном алгоритме для криптостойкости предлагается 8 раундов шифрования, в нашем модифицированном алгоритме – 63 раунда.

В алгоритме используются следующие субалгоритмы, кодовые номера которых можно менять:

Таблица 1 – Субалгоритмы HPC

NC	ОБОЗНАЧЕНИЕ	РАЗМЕР БЛОКА
9	HPC-Tiny	От 0 до 35 бит
4	HPC-Short	От 36 до 64 битов
6	HPC-Medium	От 65 до 128 битов
2	HPC-Long	От 129 до 512 битов
3	HPC-Extended	От 513 до 2048 бит

Субалгоритмы можно расширять до бесконечность, причем значения N_C устанавливаютя произвольно.

ОБОЗНАЧЕНИЯ

Обозначения, используемы в алгоритме:

- $\oplus$ - операция побитового сложения по модулю 2
- $\boxplus$ и $\boxminus$ - операции сложения и вычитания 128 битных операндов по модуля 2^{128}
- <<n – циклический сдвиг влево (или вправо) на указанное фиксированное число битов
- << - циклический сдвиг влево (или вправо) на переменное число битов, определяемое значением 6 младших битов «бокового» параметра
- t() – функция обнуления младшего байта 64-битного операнда

АЛГОРИТМЫ ШИФРОВАНИЯ

- Sec_1, Sec_2, Sec_3 – секретные ключи, вычисляются как 19-20 значные хэш суммы секретных фраз
- N_C – значение субалгоритма (см. Таблица 1)
- b – размер шифруемого блока в битах
- C_1 - константа вычисляется по формуле:
  $C_1 = b + Sec_1$
- L – размер ключа шифрования в битах
- K[] – массив расширенного ключа, инициализируется по формулам:
  $K[0] = Sec_1 + N_C$
  $K[1] = Sec_2 + L$
  $K[2] = Sec_3 << N_C$
  Остальные 509 слов массива инициализируются следующим образом:
  $K[i] = K[i-1] + (K[i-2] \oplus (K[i-3] >> 21) \oplus (K[i-3] << 37)) \bmod 2^{128}$
- SuperKey – массив дополнительного ключа, состоящий из 8-ми 128 битных слов

СТРУКТУРА РАУНДА

Предварительно шифруемые данные записываются в 128 битные регистры S_0 и S_1, над содержимым которых в каждом раунде выполняется множество элементарных операций.

X_n – значения, определяющие количество битов циклического сдвига в соответствующей операции:

$$X_1 = 22 + (<S_0>\&31)$$

$$X_2 = 33 + i$$

где:

$<S_0>$ - текущее на момент выполнения операции значение регистра S_0

& - операция побитового логического «и»

i – номер текущего раунда (начиная с 0)

k_n – фрагмент расширенного ключа. Процедура расширения ключа:

$$k_1 = K[<S_0>\&511]$$

$$k_2 = K[<S_0>\&511]$$

$$k_3 = K[<S_0>\&255 + 3*i + 1]$$

$$k_4 = K[b + 16 + i]$$

Sp_n – фрагмент дополнительного ключа, участвующего в операции шифрования

$$Sp_1 = SuperKey[i \oplus 4]$$

$$Sp_2 = SuperKey[i]$$

$$Sp_3 = SuperKey[i \oplus 7]$$

$$Sp_4 = SuperKey[i \oplus 2]$$

$$Sp_5 = SuperKey[i \oplus 1]$$

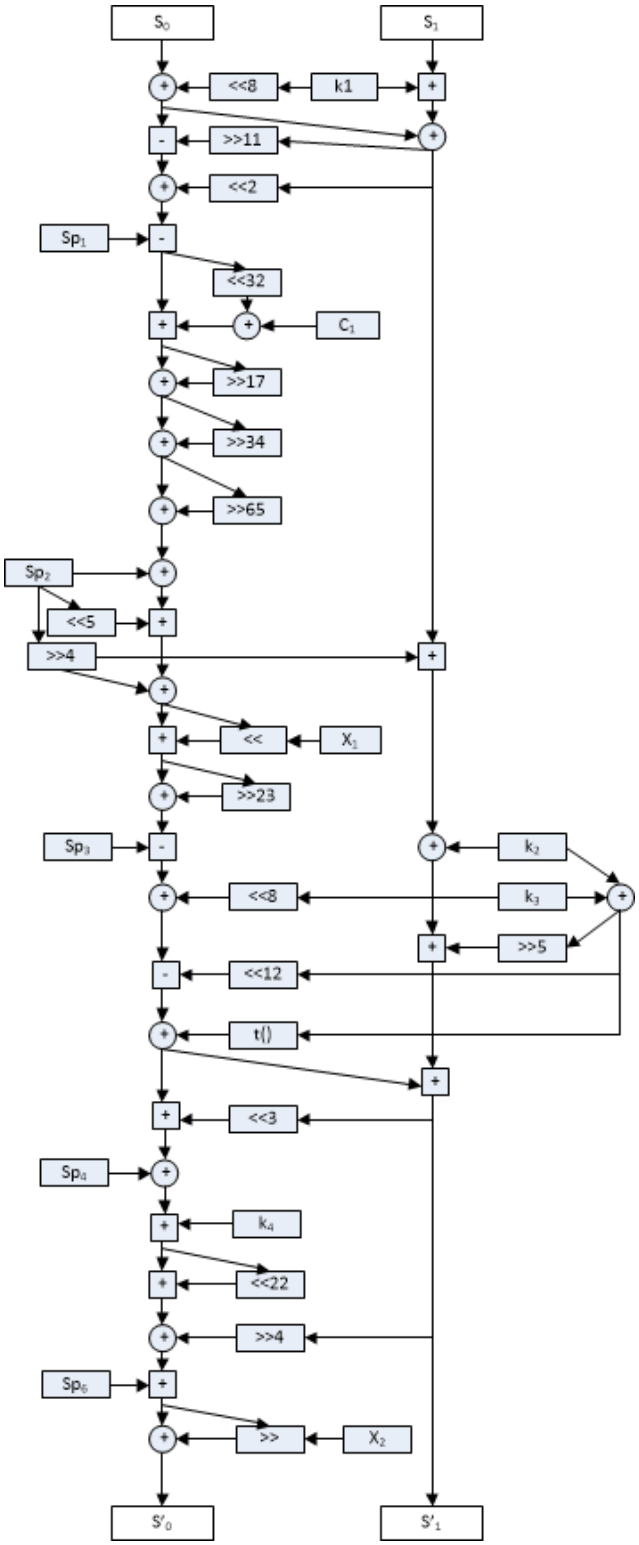

Рисунок 1 – Алгоритм раунда модифицированного HPC

АЛГОРИТМЫ ШИФРОВАНИЯ

В каждом раунде шифрования выполняются операции с 128-битными словами, т.к.алгоритм предназначен для 128-битных процессоров.

Для данного алгоритма будем применять 63 раунда. По завершении каждого раунда S_0 преобразуется в S'_0, а S_1 – в S'_1, и на вход следующего раунда подаются S'_1 в качестве S_0 и S'_0 в качестве S_1.

По завершению 63 раундов преобразований выполняются 2 дополнительные операции:

значение S'_0 (т.е. значение регистра S_0 после выполнения приведенных на рисунке 1 действий) складывается по модуля 2^{128} с K[b+8], а S'_1 – с K[b+9].

Расшифровывание производится выполнением обратных операций в обратной последовательности.

ПРОЦЕДУРА РАСШИРЕНИЯ КЛЮЧА

Задача процедуры расширения ключа – формирование расширенного ключа, представляющего собой массив из 512 128-битных слов. Причем для каждого из субалгоритмов должен быть отдельный массив расширенного ключа (процесс расширения ключа для каждого субалгоритма различен). Преимущество алгоритма в том, что знание одного из массивов расширенного ключа не позволяет вычислить значения других расширенных ключей, а также исходный ключ шифрования. Алгоритм шифрации использует переменный размер блока шифруемых данных от меньшего значения N_c к большему и так по кругу : 24 блока HPC-Long (N_c=2), затем 24 блока HPC-Extended (N_c=3) и т.д. Число 24 связано с особенностями потокового мультипроцессора, в дальнейшем это число будет увеличиваться, что положительно будет сказываться на скорости шифрации.

Ниже приведены шаги алгоритма:

1. Инициализация массива расширенного ключа K[]
2. Производится побитовое сложение по модулю 2 ключа шифрования и проинициализированного массива расширенного ключа.
3. Выполняется функция перемешивания данных расширенного ключа, которая обеспечивает влияние каждого бита ключа шифрования на значение каждого бита расширенного ключа. Далее приведена последовательность операций функции перемешивания.
    a. Выполняется инициализация регистров S0…S7 следующим образом:
    S7=K[511],
    S6=K[510],
    S5=K[509],
    S4=K[508],
    S3=K[507],
    S2=K[506],
    S1=K[505],
    S0=K[504]

b. Для каждого слова расширенного ключа производятся вычисления, приведенные на рисунке 2, причем для усиления эфекта перемешивания этот этап выполняем в 9 раундов перемешивания (автор алгоритма рекомендует 3 раунда перемешивания).
c. На рисунке 2 мсользованы следующие обозначения:
    i. | - операция побитового логического “или”
    ii. & - операция логического и по модулю 2
    iii. i – номер вычисляемого слова расширенного ключа
    iv. j – номер раунда выполнения этапа 3
    v. kc_n – текущие значения слов расширенного ключа, выбираемых определенным образом:
    $kc_1 = K[i]$
    $kc_2 = K[(i+83)\&511]$
    $kc_3 = K[<S_0>\&255]$

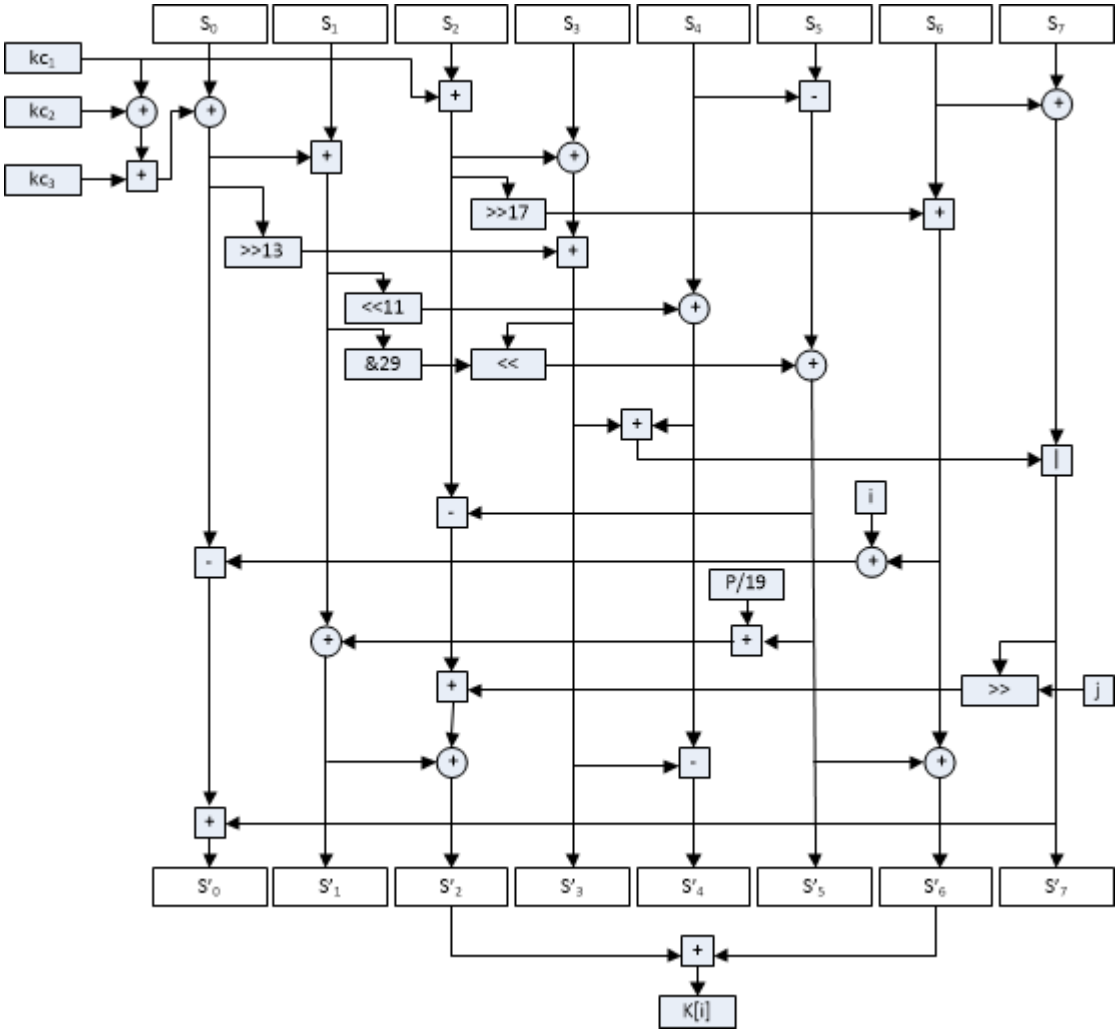

Рисунок 2 – Алгоритм раунда расширения ключа

4. Оптимальный максимальный размер ключа шифрования 65536 бит (64кБ), что является более чем достаточным размером с учетом того, что на рынке на данный момент присутствуют решения с килобитным максимальным ключом. 64кБ - оптимальный максимум для выбранного процессора, но 64 кБ не является предельным максимальным значением, можно и больше, тогда ключ разбивается на 64кБ фрагменты. Для ключа, либо фрагментов, загруженных в локальную память потокового процессора производится логическое разбиение на 128 словные фрагменты для которых повторяются шаги 2 и 3.

ДОСТОИНСТВА И НЕДОСТАТКИ АЛГОРИТМА

Раунд алгоритма HPC состоит из огромного количества операций. В сравнении, например, с раундом стандарта симметричного шифрования ГОСТ 28147-89 алгоритм HPC выглядит чрезвычайно сложным. Тем не менее, поскольку подавляющее большинство действий выполняется с 128 битными операндами на 128 битном процессоре ожидается большая производительность, причем с учетом того, что на рынке в основном присутствуют 64 битные процессоры общего назначения это накладывает определенные сложности при взломе. При анализе криптостойкости не были выявлены слабые стороны, однако сложность алгоритма и последующая ее модернизация теоретически дает поле для творчества, но увеличенное число раундов и битность, 64 килобитный ключ и неизвестность алгоритма для взломщика позволяют считать алгоритм криптостойким. Другим недостатком алгоритма является медленная процедура расширения ключа шифрования, которая не может выполняться параллельно с самим шифрованием данных, что потребует некоторого изящества (а значит времени) при оптимизации. К достоинствам алгоритма стоит отнести и тот факт, что он был отвергнут Американским правительством, являясь при этом более криптостойким чем AES.

Модифицированный DFC(серверное усиление шифрации)

ОПИСАНИЕ АЛГОРИТМА

DFC – Decorrelated Fast Cipher быстрый шифр без взаимосвязей. Серверное усиление предполагается при использовании двойной шифрации HPC(DFC) алгоритмами. Усиление предполагает взлом операционной системы сервера и дискредитацию ключей либо бинарного кода HPC алгоритма. Также предполагается хранить ключи, шифры, исходники, бинарные коды на сервере под этим усилением.

Модифицированный E2 (мобильное усиление шифрации)

ОПИСАНИЕ АЛГОРИТМА

Алгоритм E2 – алгоритм симметричного шифрования на основе сети Фейстеля. Мобильное усиление предполагается при использовании двойной шифрации HPC(E2) алгоритмами. Усиление предполагает факт дискредитации ключей и бинарного кода алгоритма HPC.

Модифицированный CAST-256(усиление шифрации для энергонезависимой памяти)

ОПИСАНИЕ АЛГОРИТМА

Алгоритм CAST-256 – алгоритм симметричного шифрования на основе сети Фейстеля с фиксированным размером ключей шифрования: 128, 160, 192, 224 и 256 битов. Усиление предполагается при использовании двойной шифрации HPC(CAST-256) алгоритмами. Усиление предполагает факт утери зашифрованных данных вместе с ключами и бинарным кодом шифра на одном носителе информации и тем самым временную дискредитацию HPC алгоритма

Апаратная реализация

АППАРАТНАЯ ЧАСТЬ ДЛЯ РЕАЛИЗАЦИИ АЛГОРИТМОВ ШИФРАЦИИ

Реализация алгоритмов возможна на процессорах nVidia с поддержкой технологии CUDA.

Полный список: (https://developer.nvidia.com/cuda-gpus)

Бюджетные карты для ноутбуков и десктоп компьютеров:
(http://www.nvidia.com/object/geforce_family.html - полный список).

Мобильные решения на базе процессора NVIDIA Tegra 4 с Android OS:

Смартфоны: http://www.nvidia.com/object/tegra-superphones.html

Планшетники: http://www.nvidia.com/object/tegra-supertablets.html

Для сервера с высокой нагрузкой можно будет применить

ПАРАМЕТРЫ ВЫЧИСЛИТЕЛЯ

Первоначальная реализация алгоритмов выполняется на видеокарте GeForce GT 440 со следующими параметрами:

ПАРАМЕТР	ЗНАЧЕНИЕ
Compute capability	2.1
Name	GeForce GT 440
Total Global Memory	1073741824
Total Constant Memory	65536
Shared Memory per block	49152
Registers per block	32768
Warp size	32
Max threads per block	1024
Clock Rate	1620000
Multiprocessor Count	2
Total Processors	2x2x24=96
Max Threads Dim	1024 1024 64
Max Grid Size	65535 65535 65535

ВЕРСИЯ CUDA

На момент написания используется версия 4.2 (5.0 уже вышла но пока есть сложности с установкой)

АПАРАТНАЯ РЕАЛИЗАЦИЯ

ПРИМЕРЫ НОУТБУКОВ ПОДДЕРЖИВАЮЩИХ CUDA

Ноутбук MSI GT70 ON - 59 990 рублей

Core i7 3630QM 2.4 ГГц (в режиме Turbo Boost 3.3 ГГц) и супермощной дискретной видеокарты GeForce GTX 670MX с 3Гб памяти

CUDA 960 cores

http://www.digital.ru/goods/notebooks/msi/708525.htm

Sony S1512X1RB - 56 490 рублей

Intel Core i7

GeForce® GT 640M LE

CUDA 384 cores

http://www.digital.ru/goods/notebooks/sony/709024.htm

SONY SVS1512U1R 45 990 рублей

Intel Core i5-3210M

GeForce® GT 640M LE

CUDA 384 cores

http://www.digital.ru/goods/notebooks/sony/710072.htm

NVIDIA GeForce GT 640M - 25 990 рублей

Intel Core i5 3210M

GeForce® GT 640M

CUDA 384 cores

http://www.digital.ru/goods/notebooks/acer/709662.htm

Варианты использования

СПИСОК АКТЕРОВ

АКТЕР	ОПИСАНИЕ
Властелин колец	Генерация алгоритма шифрования, изменение параметров для генерации криптомодуля
Доверенные лица	Ввод секретных слов по которым строится алгоритм, секретные слова никогда не должны повторяться
Разработчик	Разработка алгоритмов шифрации и скриптов генерации
Системный администратор	Поддерживает актуальность криптомодулей, генерирует ключи шифрования, настраивает взаимодействие крипто сервера с роутером, отслеживает работоспособность системы и оповещает Властелина Колец в случае подозрительной сетевой активности

СПИСОК СИСТЕМ

НАИМЕНОВАНИЕ	ОПИСАНИЕ
Router	Сетевое оборудование
CPU Crypto Server	Сервер шифрования
GPU Crypto Server	Вычислитель сервера шифрования
CPU Crypto Key Generator	Обычгый компьютер

СПИСОК ПРОГРАММ

ПРОГРАММА	ОПИСАНИЕ
Генератор криптографического модуля	Позволяет генерировать криптографические бинарные модули
Веб интерфейс ввода секретных фраз	Простой вебинтерфейс для ввода секретных фраз, работает кратковременно и только в защищенной сети
Генератор ключей	Генреатор ключей, программа для системного администратора
Настройка крипто сервера	Настройка крипто сервера для системного администратора

ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ

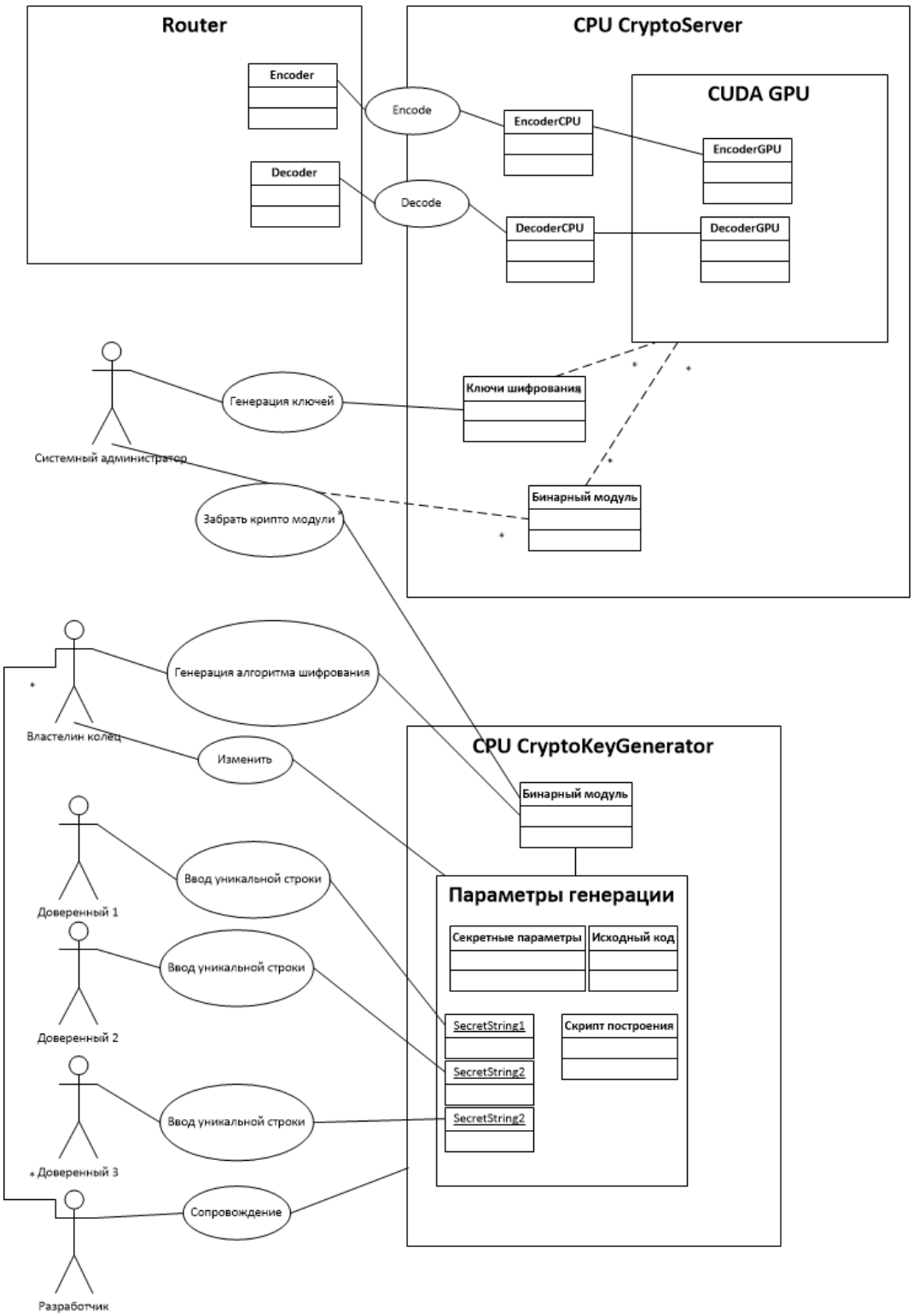

Рисунок 3 – Варианты использования системы

ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ

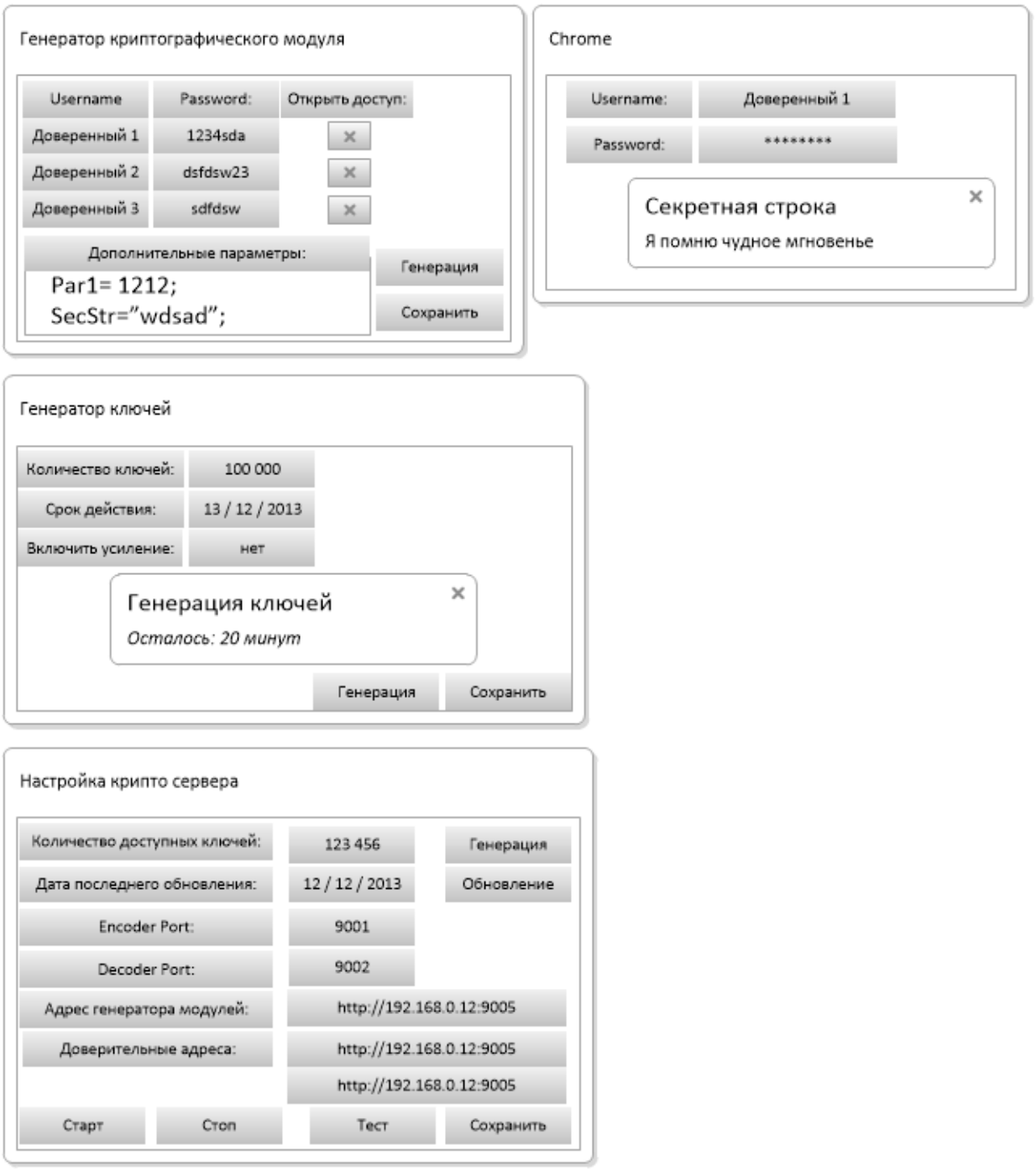

Рисунок 4 – Макеты приложений

ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ

ОСНОВНЫЕ ПРОГРАММНЫЕ СУЩНОСТИ

ИМЯ	ОПИСАНИЕ
SecretString	Секретные строки по которым вычисляются хэш суммы, используемые в генерации кода
Исходный код модуля	Код CUDA модуля
Скрипт построения модуля	Скрипт для компиляции и построения бинарного CUDA ядра
Секретные параметры для построения	Параметры скрипта построения
Бинарный модуль	CUDA ядра – бинарные 128битные модули шифрования
Ключи шифрования	Ключи
DecoderGPU	Программа декодирования, работающая на CUDA вычислителе
EncoderGPU	Программа кодирования, работающая на CUDA вычислителе
DecoderCPU	Программа декодирования, работающая на CPU крипто сервера
EncoderCPU	Программа кодирования, работающая на CPU крипто сервера
Decoder	Модуль декодирования на сетевом устройстве
Encoder	Модуль кодирования на сетевом устройстве