Код Безопасности

конференция РусКрипто

# Эффективная реализация алгоритма ГОСТ 28147-89 с помощью технологии GPGPU

Алексей Кролевецкий
Ведущий программист

# Описание проблематики

- Современное развитие ИТ-инфраструктур требует высоких скоростей шифрования данных ~10 Гбит/с.
- Необходимо повышать скорость шифрования ГОСТ 28147-89
- Это возможно за счёт использования:
  - Параллелизма при обработке данных
  - Алгоритмических усовершенствований
  - Использования аппаратных возможностей

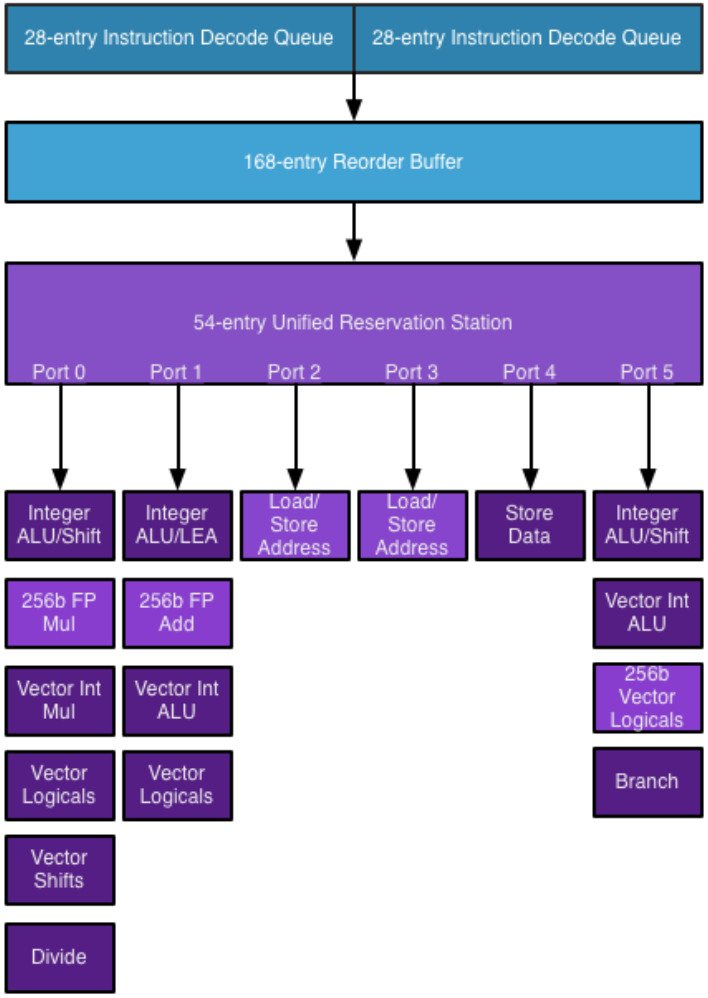
# Шифрование на CPU

- Архитектура x86/x64 с расширениями SIMD
- Шифрование на регистрах общего назначения
  - Использование таблиц S-Box размером 4 Кбайт
  - Скорость: ~60 тактов/байт,
  - Потоки данных: 1 поток/ядро
- Шифрование на регистрах SSE 128 бит
  - Использование SIMD команд SSE для параллельной обработки нескольких потоков данных
  - Использование AVX для уменьшения операций по пересылке данных
  - Скорость ~8 тактов/байт,
  - Потоки данных: 16 потоков/ядро

# Операция подстановки на x86/x64

- Аппаратная реализация
  - Команда PSHUFB из расширения SSSE3
    - Появилась в процессорах AMD начиная с Bulldozer (2011 г.)
- Программная реализация
  - С помощью таблиц подстановки
    - Обращение к памяти – random
    - Скорость зависит от количества блоков LD процессора
    - Таблица должна помещаться в кэш ЦП
  - Операция подстановки как полином
    - Обращение к памяти – линейное
    - Таблица коэффициентов полинома – 64 байт
    - Параллельная обработка 8-и тетрад
    - Параллельный расчет самого полинома

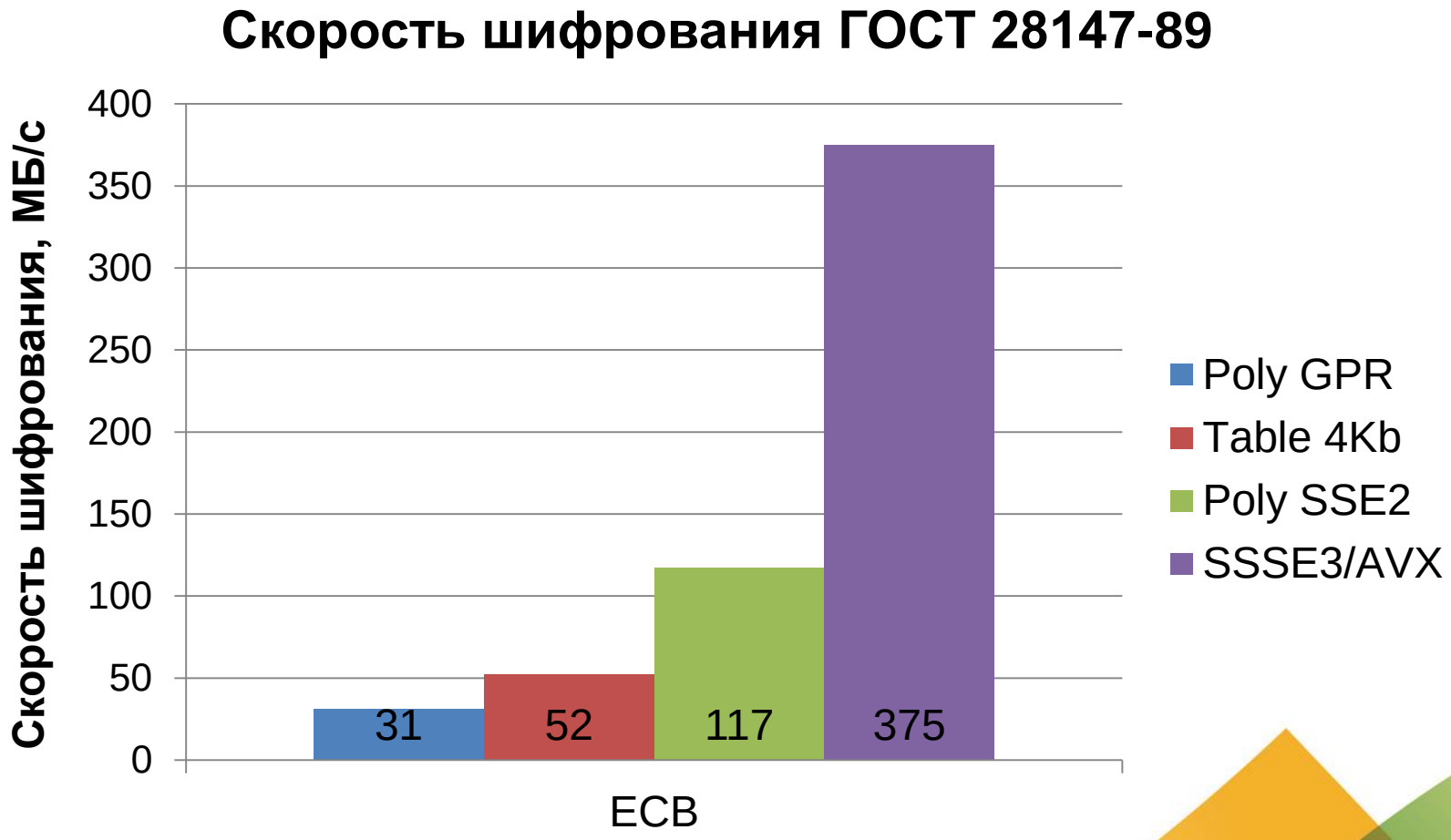
# Микроархитектура CPU




- В зависимости от архитектуры ЦП меняется скорость шифрования
- 2 блока Load Data
- 3 блока исполнения инструкций
  - Параллельное исполнение
    - 3-х ALU/SSE операций
    - 2-х команд PSHUFB

# Скорость шифрования CPU

- Процессор Intel Core i5-3570 3,4 ГГц 1 поток
  - Технологии Turbo Boost и SpeedStep отключены



# Производительность

- Целевая система для шифрования использует только процессоры
- Производительность такой системы растет пропорционально:
  - Числу ядер в системе
  - Рабочей частоте ядер
- Производительность платформы АПКШ «Континент» IPC-3000F может достигать ~20 Гбит/сек при параллельной обработке более 256 потоков шифрования

# GPGPU

- Для повышения скорости шифрования платформы предлагается использовать GPU
- GPGPU – техника использования GPU для вычислений общего назначения
- GPU обладает большим числом ядер чем CPU
- Рабочая частота GPU ~1ГГц (в 2-3 раза меньше частоты CPU)
- Система памяти GPU ориентирована на большую пропускную способность при больших задержках
- Ограниченность ресурсов ядер GPU

# Реализации GPGPU

- CUDA – только GPU Nvidia
- OpenCL – поддерживает различные устройства
- DirectCompute – только OS Windows
- C++ AMP - только OS Windows
- OpenACC – надстройка над CUDA и OpenCL
- AMD CAL – только GPU AMD

# OpenCL

- OpenCL – открытый стандарт параллельных вычислений на GPU, CPU, FPGA и DSP
- Преимущества:
  - Кроссплатформенность
  - Поддержка GPU от разных производителей
  - Событийно-ориентированное программирование
  - Одновременное вычисление и передача данных
- Недостатки:
  - Нет параллельного исполнения различных kernels
  - Нет ассемблера для GPU

OpenCL

# Тестовые GPU

- Характеристики исследованных GPU

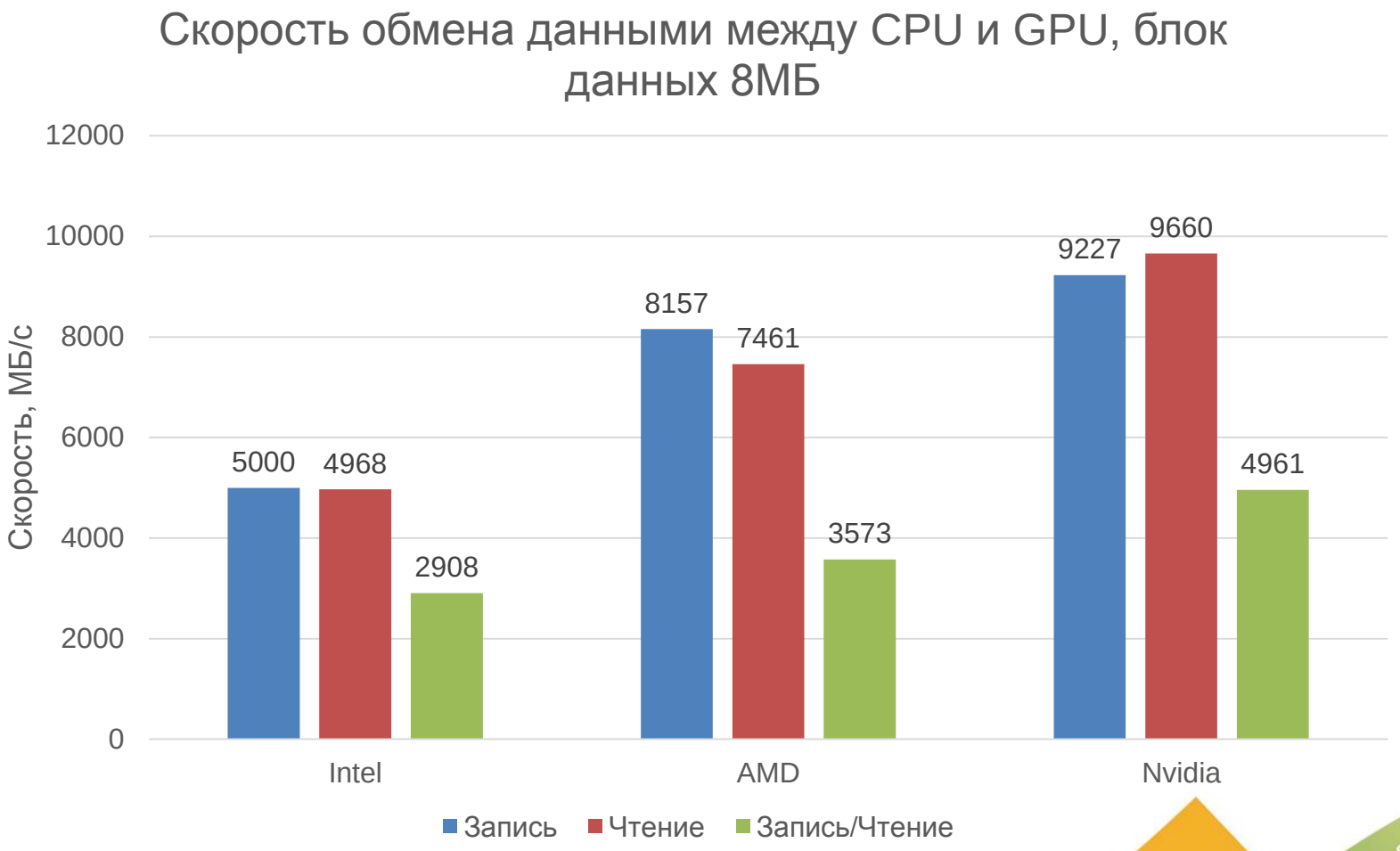
| Модель | Intel HD 2500 | Nvidia GeForce GTX 750 | AMD HD 7790 |
|---|---|---|---|
| Архитектура | GT1 | Maxwell | GCN 1.1 |
| Кол-во ядер, шт. | 6 | 16 | 14 |
| ALU/ядро, шт. | 4×3 | 32 | 64 |
| ALU всего, шт. | 72 | 512 | 896 |
| Частота, МГц | 1150 | 1250 | 1075 |
| Шина | Ring bus | PCIe 3.0 | PCIe 3.0 |

# Шифрование на GPU

- Используем метод SPMD для достижения параллелизма
- Для эффективности необходимо шифровать параллельно ~10000 потоков
- Для каждого потока используется свой ключ
- Event-driven FSM
- Накладные расходы связанные с передачей данных в GPU необходимо минимизировать:
  - Использовать pinned memory и DMA
  - Двойная буферизация данных

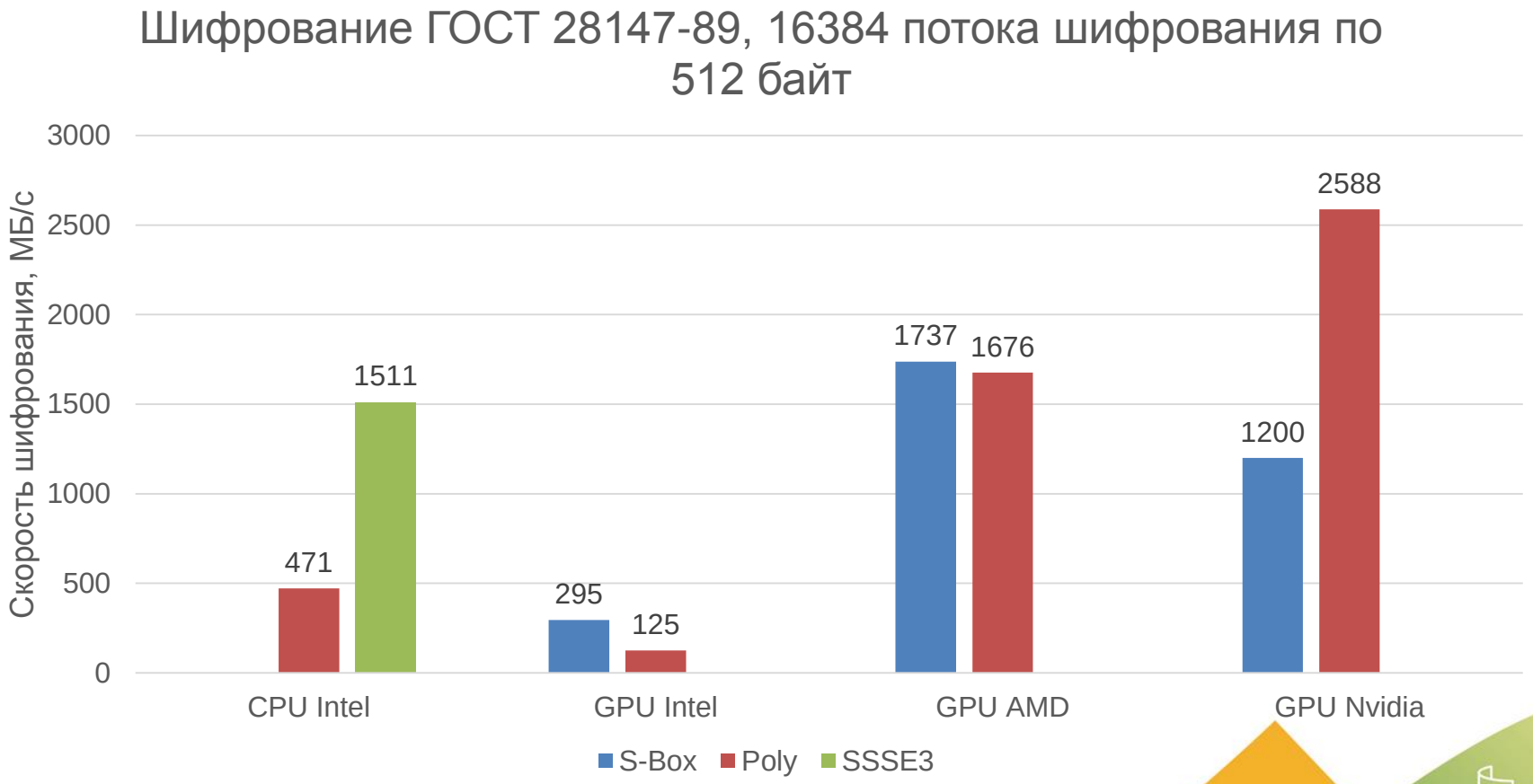
# Передача данных

- Измерение скорости PCIe 3.0, OpenCL



# Шифрование GPU

- Шифрование OpenCL, блок 8 МБ, с учетом передачи данных

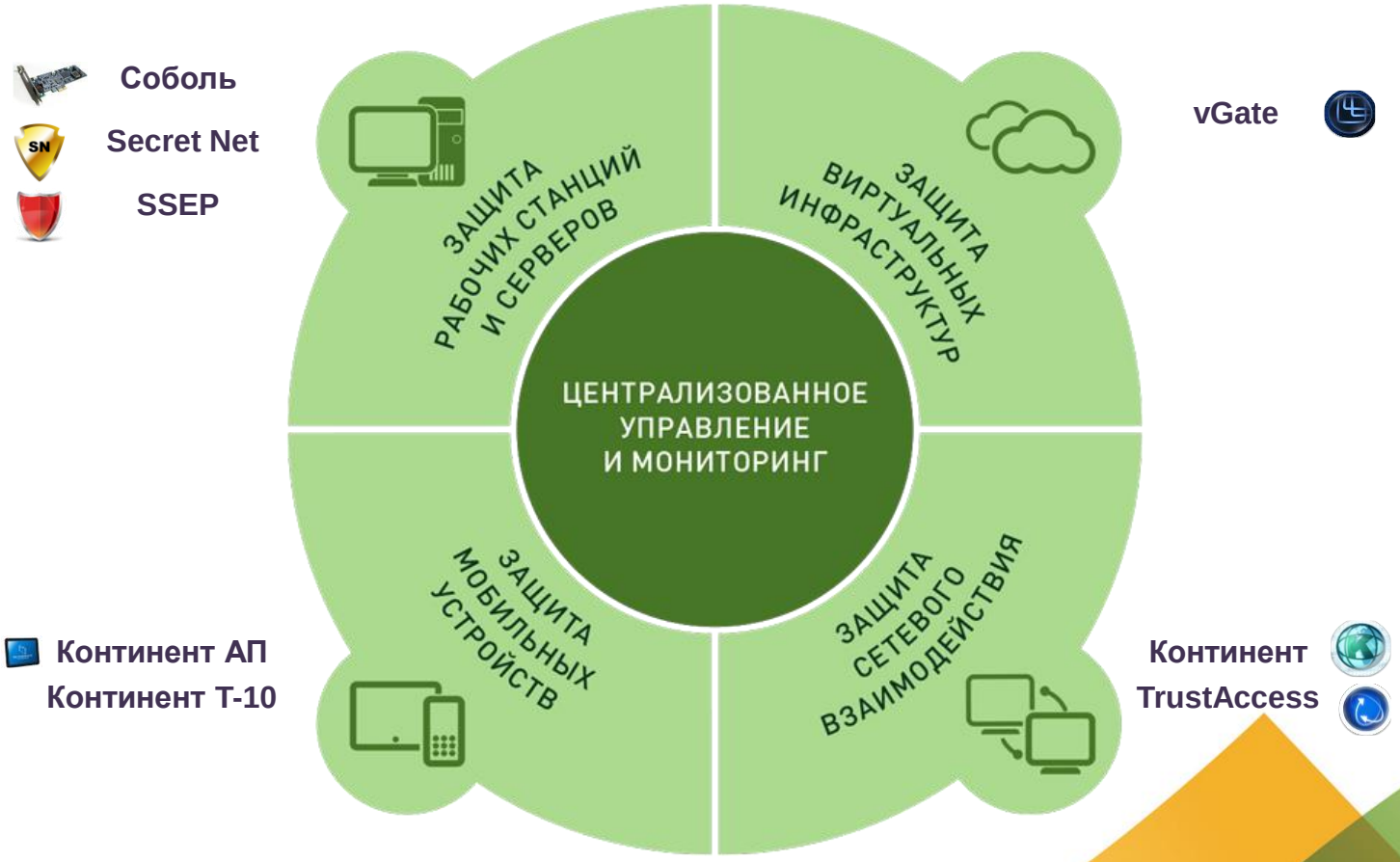


# Выводы

- Шифрование на GPU позволяет увеличить производительность целевой платформы в 2 раза
- Пропускная способность PCIe 3.0 не является узким пестом при реализации шифрования с помощью GPU
- Для каждой модели GPU эффективен свой алгоритм шифрования

# О «Коде Безопасности»

- Разработчик широкой линейки продуктов для управления доступом и защиты информации в соответствии с требованиями регуляторов




Код Безопасности

# Вопросы?

Код Безопасности

Алексей Кролевецкий

a.krolevetsky@securitycode.ru