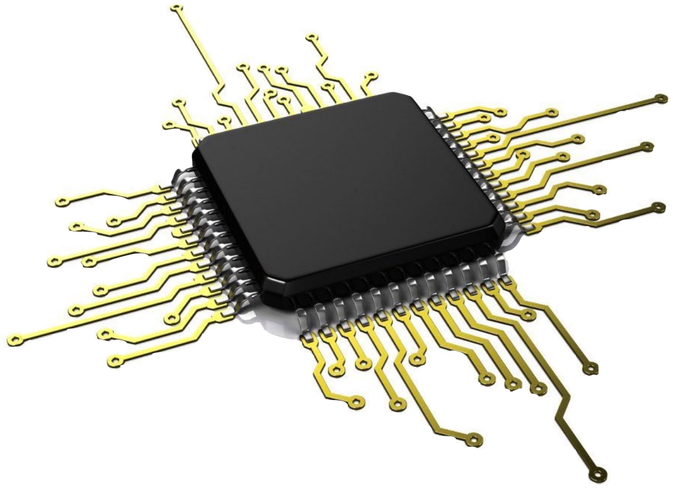
Российская элементная база и криптография для обеспечения доверия и безопасности IIoT-устройств для объектов КИИ



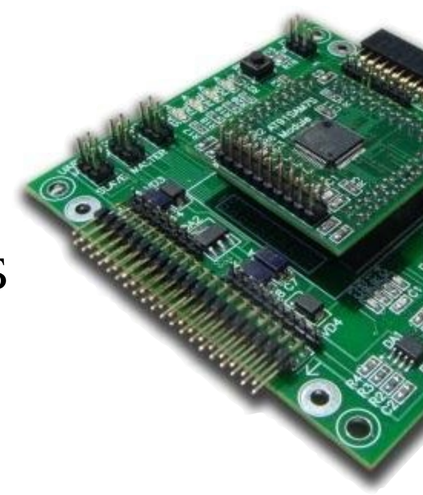
Сергей Груздев

ген. директор

"Аладдин Р.Д."

Типовые требования по безопасности IIoT и M2M-устройств

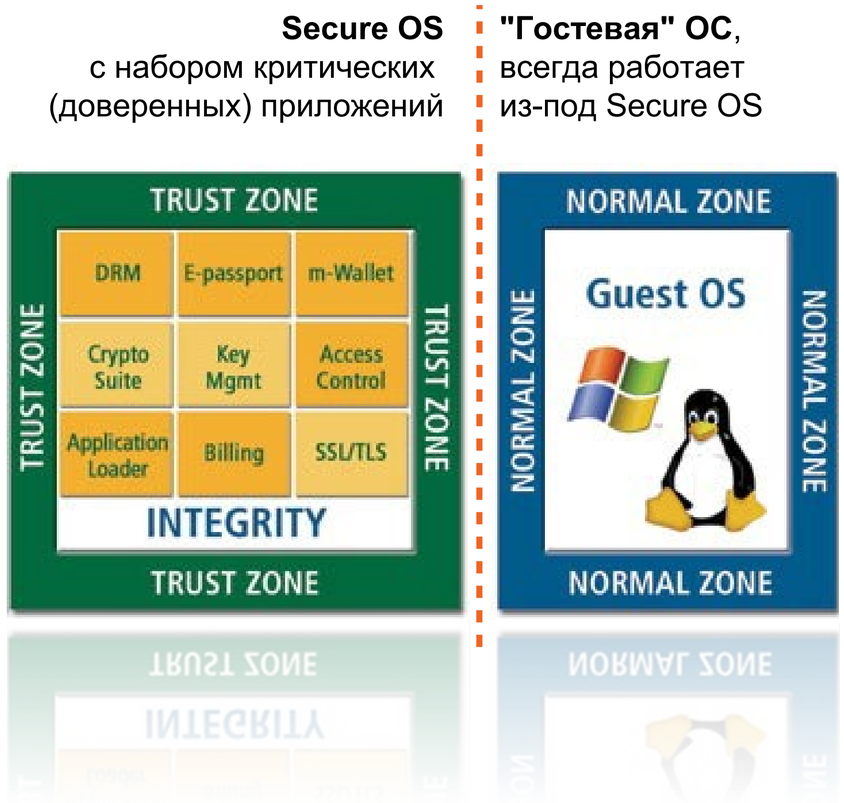
- Что сегодня прописывают в Требованиях?
  - Смотрим типовую конкурсную документацию
    - Некорректируемость данных (счётчики, трекеры, навигация…)
    - Идентификация/Аутентификация (чего/кого??? понимают как проверку серийного номера)
    - Защита передаваемых данных с использованием ГОСТовых алгоритмов
- Этого достаточно для безопасности IoT-устройств и создаваемой инфраструктуры?
  - Категорически нет
  - Для этого нужен PKI? Ответ: нет
- Продвинутые требования (когда подключаются ИБ-компании)
  - Добавляется "Secure Element" (доп. чип, модуль), реализующий несколько функций ИБ (аутентификация, ЭП, шифрование, защита канала,..)
  - Зачем?
    - Отстроиться от конкурентов (защита сделок)
  - Чуть лучше реализовать Требования (с использованием российских СКЗИ) - всё те же наложенные средства



✓ Задача обеспечения безопасности IoT-устройств и инфраструктуры не решается

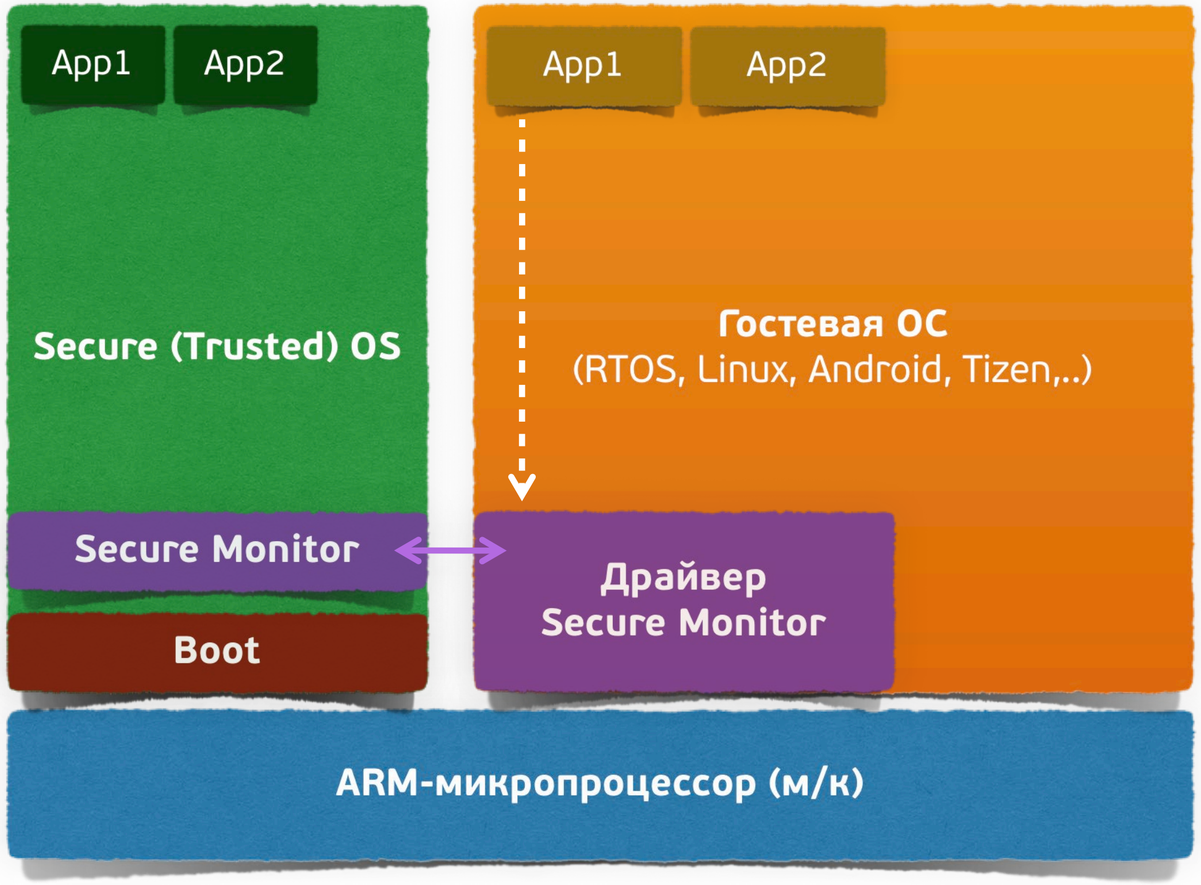
На чём делаются IIoT и M2M-устройства

- Большинство современных устройств построено на базе микропроцессоров архитектуры ARM (System-on-Chip - SoC)
  - Начиная с ARM R4 (для промышленной электроники и встраиваемых систем) и Cortex A5 (мультимедийные процессоры) и в микропроцессорах появилась т.н. TrustZone
  - Технология TrustZone была стандартизирована в 2010 г. и легла в основу Device GlobalPlatform
- TrustZone - аппаратная изоляция (виртуализация) двух параллельных процессов ("миров")
  - "Доверенного" / безопасного (Secure World)
  - Нормального / обычного (Normal World)
    - Здесь работают приложения под управлением "гостевых" ОС - RTOS, Linux, Android, Tizen, Sailfish, Аврора…



Архитектура современных ARM-процессоров

Управление обменом с периферией производится через TrustZone и встроенные контроллеры



#1 - Загрузчик

- получает управление сразу после включения питания, стартует Secure OS

#2 - Secure OS загружает "гостевую" ОС

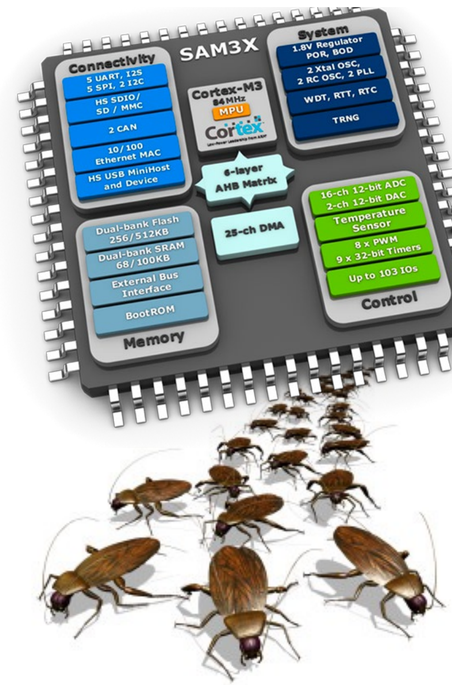
#3 - Secure OS может контролировать всё

- "Гостевая" ОС и установленные в ней приложения узнать об этом не смогут

Архитектура современных ARM-процессоров

- Практически все ARM-процессоры и м/к выпускаются с закрытой TrustZone
  - Приложения из TrustZone может скрытно и необнаруживаемо выполнять шпионские функции
    - Контролировать обмен данными, манипулировать информацией (подменять GPS/Глонасс координаты и пр.), красть криптографические ключи и т.п.
    - Требование PPD-20/2012 (Б. Обама) ко всем американским вендорам
      - Скрытый сбор (встраивание закладок) и передача в АНБ, ФБР, ЦРУ "телеметрии"
      - Запрет на раскрытие исходных кодов
  - При инициализации м/к ядро Linux обращается к механизмам TrustZone
    - Secure Monitor Call - внешне ничем не примечательная команда SMC #0
    - "Движки" большинства ОС (Android, Tizen, Sailfish) построены на ядре Linux
    - Без этого ядро просто не запустится (инициализация кэш L2, управление питанием и пр.)

✓ TrustZone - это не место в контроллере, его нельзя выкусить или обойти!



- Процессоры без TrustZone не безопасны и уязвимы

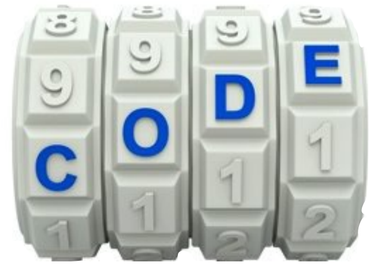
Слагаемые безопасности IoT-устройств (1)

1. Выбор правильной элементной базы

- Собрали БД на чём нельзя делать - более 25К клонируемых, уязвимых, взломанных м/к (НИР для ФСТЭК)

2. Secure by design

- Правильная методология проектирования, разработки, тестирования, производства и эксплуатации IoT-устройств



3. Доверенная загрузка (МДЗ)

- Первая исполняемая команда при включении питания м/к должна быть наша
- Предзагрузчик (pre-boot код SCP-контроллера) должен быть "зашит" в ROM-процессора
- При активации МДЗ в процессоре должны пережигаться перемычки (eFuse) и прошиваться MasterKey процессора, на котором будет зашифрован исполняемый код
- Код МДЗ должен быть зашифрован и подписан, проверка ЭП и расшифрование кода кода МДЗ должны производиться встроенным в процессор криптографическим модулем

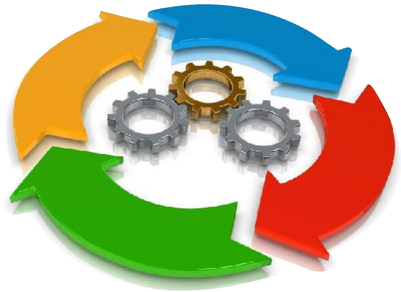
4. Исполнение ТОЛЬКО доверенного кода

- Исполняемый код должен быть подписан и зашифрован

Слагаемые безопасности IoT-устройств (2)

5. Безопасность кода

- Разработка встраиваемого ПО должна вестись в соответствии с требованиями SDL (Разработка безопасного ПО - ГОСТ Р 56939-2016)
  - Код должен быть тщательно протестирован - отсутствие ошибок, предупреждений ("грязный" код), уязвимостей
  - Качество автотестов должно проверяться с помощью мутационного тестирования (изменение одной команды или бита в тестируемом коде должно обнаруживаться, иначе - это плохой автотест)
  - Для качественного тестирования и фаззинга нужен программный эмулятор процессора
  - Код должен быть написан с учётом особенностей h/w и правил Security-программирования (исключение однобитовых условий, учёт нагрузки на EEPROM/Flash-память, сборка ключей шифрования только на регистрах процессора и т.п.)



6. Возможность безопасного дистанционного обновления "прошивки"

- Никто не научился писать код без ошибок - его нужно будет обновлять
- Нельзя бездумно использовать общеизвестные протоколы (например, DFU - он содержит заложенные уязвимости) - нужно всё внимат ельно анализироват ь

Слагаемые безопасности IoT-устройств (3)

7. Встроенная Secure OS

- Должна контролировать все каналы "общения" м/к с внешним миром (помним, что это SoC - System On Chip), инициировать только используемые контролеры, интерфейсы и протоколы
- Должна контролировать распределение памяти, всех аппаратных ресурсов, работу "гостевой" ОС и её общение с "внешним" миром

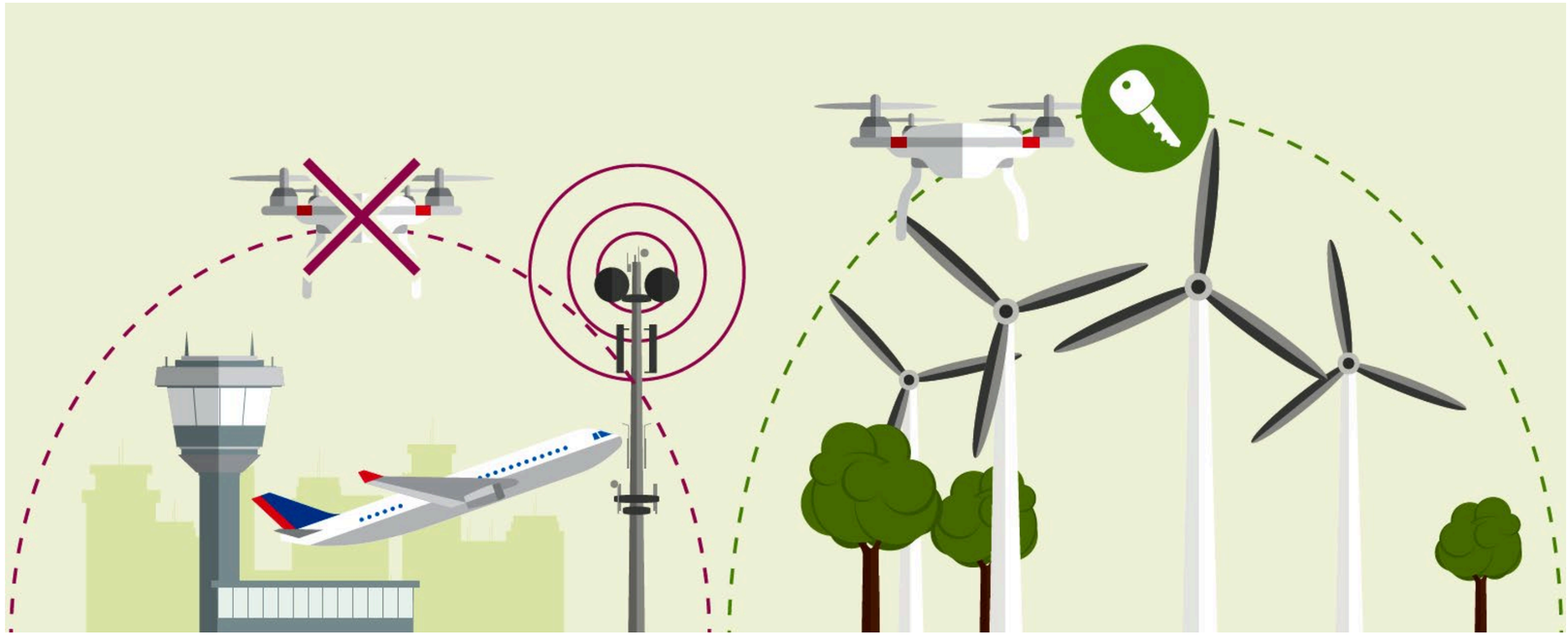
8. Изолирование критически важных приложений от "гостевой" ОС

- Критически важные для безопасности IoT-устройств приложения
  - СКЗИ (генерация ключей, шифрование, хэш, ЭП с неизвлекаемым закрытым ключом…)
  - Дистанционное обновление прошивки, "гостевой" ОС, приложений, данных
  - Безопасное дистанционное администрирование и управление ЖЦ устройства
  - Обработка GPS/ГЛОНАСС и др.
- Критически важные приложения должны
  - Оформляться как трастлеты - подписанные доверенные приложения для Secure OS
  - Исполняться в изолированной среде (TrustZone) в привилегированном режиме
  - Общаться с "гостевой" ОС и с внешними приложениями только по "белому" списку команд и данных через Secure Monitor

Слагаемые безопасности IoT-устройств (4)

9. "Полицейский" режим

- Не везде нужен
- "Полицейский"/антитеррористический режим (например, экстренное выключение, блокирование, перехват управления и экстренная посадка беспилотника и пр.)
- Должен делаться с использованием отдельного сертификата и подписанных команд



Слагаемые безопасности IoT-устройств (4)

10. Однозначная идентификация и взаимная СТРОГАЯ аутентификация IoT-устройств и хоста (сервера)

- Реализуется ТОЛЬКО с использованием криптографии и PKI

11. Защищённый канал передачи команд и данных

- Должен строиться с использованием PKI и неизвлекаемого закрытого ключа

12. Защищённое хранилище для данных

- Должно строиться с использованием PKI и неизвлекаемого закрытого ключа

13. Безопасное дистанционное администрирование

14. Централизованное управление жизненным циклом

- Активация IoT-устройства (для безопасной транспортировки) - вывод из состояния "кирпич"
- Ввод в эксплуатацию (профилирование, настройка и пр.)
- Перевыпуск, отзыв сертификата, блокирование работы
- Учёт устройств, их паспортизация, локация, номера версий h/w и f/w и пр.
- Вывод из эксплуатации

Для реализации всего этого
и обеспечения безопасности IIoT-устройств
нужен PKI

Потянет ли PKI современная элементная база?
Да

Можно ли выполнить все эти условия без правильной доверенной российской элементной базы?

Нет

= Наш опыт работы с правильной элементной базой

Аладдин

Что есть из правильной российской ЭКБ для IoT и M2M-устройств

Конечные IoT-устройства начального уровня	Промышленная автоматика, хабы-концентраторы	Application-процессоры	
		Байкал-М (BE-M1000) - Cortex A57 (8 ядер, TrustZone) - *Делаем встроенный МДЗ + PKI* - *Все функции ИБ "из коробки" + централизованное управление*	**Байкал-С** - Cortex A75 (48 ядер, TrustZone) - *Делаем встроенный МДЗ + PKI*
"ElIoT" (1892BM268) - Cortex M33 (2 ядра, TrustZone) - Низкое энергопотребление - Встроенная поддержка PKI - Линейка готовых модулей для IoT, SDK	**i.MX6 (NXP*)** - Cortex A7 (1/2/4 ядра,TrustZone) - *Отработали технологию "оправославливания", МДЗ (сертификат ФСТЭК, до СС)* - *Встроенная поддержка PKI*		
	Российских аналогов в этом ценовом диапазоне пока нет		

С чем мы работаем

Спасибо!

Сергей Груздев
ген. директор
АО "Аладдин"
8 (985) 762 2855
www.aladdin.ru