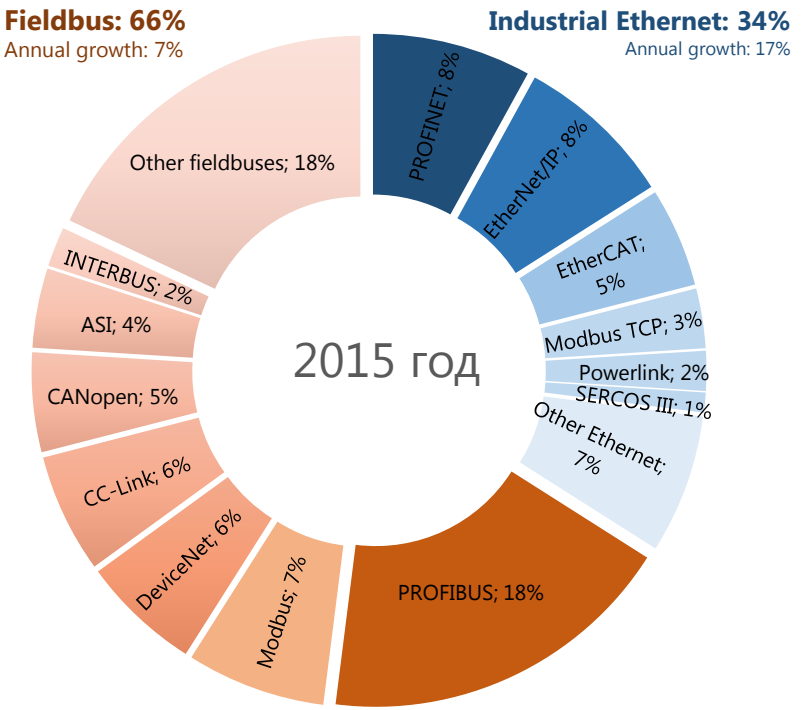
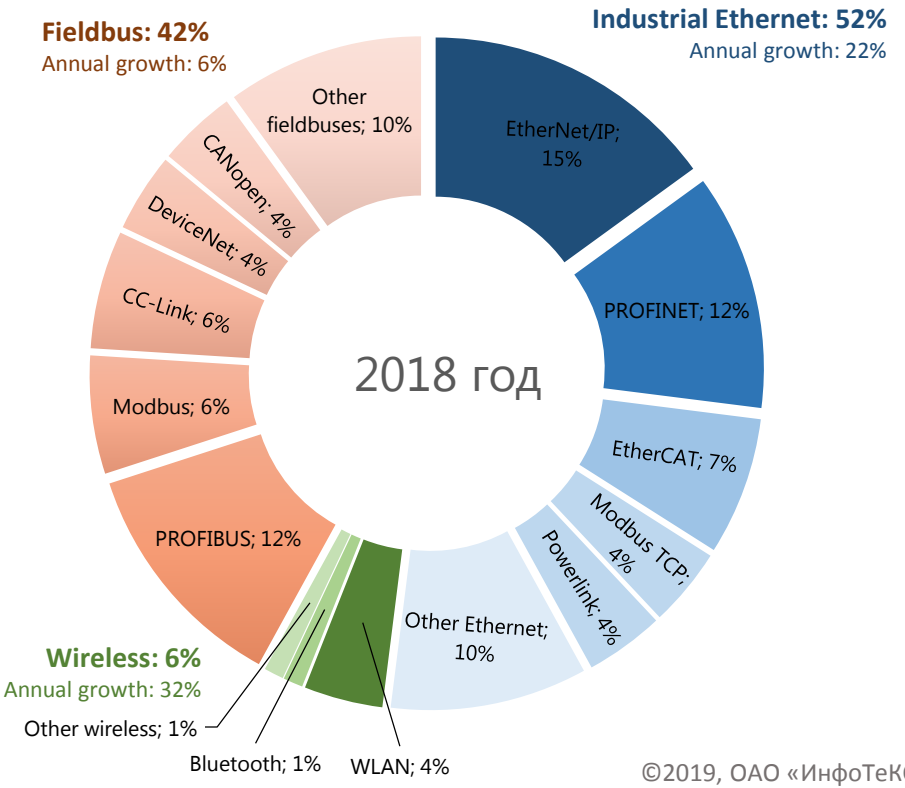
Протокол защищенного обмена для индустриальных систем (CRISP 1.0)

Шемякина Ольга,
системный аналитик ОАО «ИнфоТеКС»

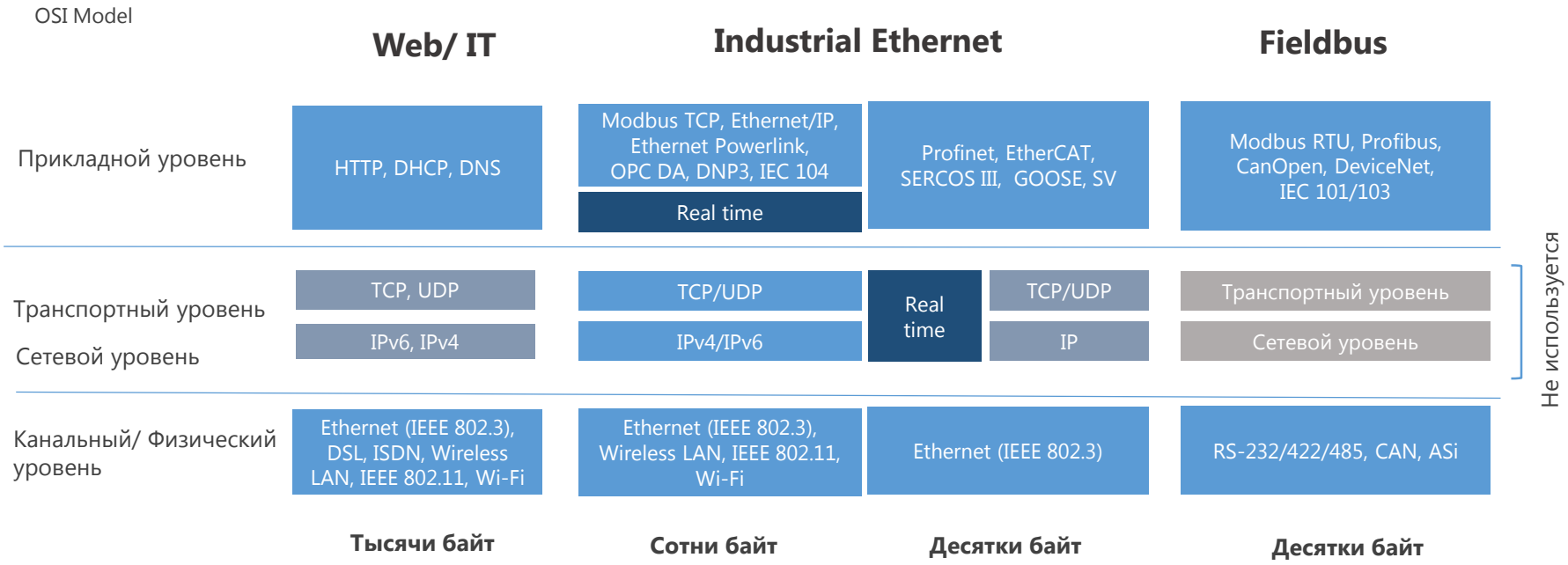
Промышленные M2M протоколы





Source: HMS Industrial Networks

Стек M2M протоколов



Особенности ИБ для M2M и IoT

Приоритеты

Web/IT

- Конфиденциальность
- Целостность
- Доступность

M2M
IoT/IIoT

- Доступность
- Целостность
- Аутентичность
- Конфиденциальность

Основные аспекты защиты M2M и IoT/IIoT коммуникаций

- Большое разнообразие протоколов
- Использование разных каналов / Использование слабых каналах
- Распространенность мультикаста и подписочной модели
- Многие протоколы являются real-time и критичны к задержкам
- Передача данных объемом в десятки-сотни байт/ критичность к оверхеду
- Большая часть M2M протоколов не являются TCP/IP base
- M2M протоколы в большинстве не подразумевают механизмов защиты коммуникаций
- Аутентичность и целостность важнее конфиденциальности

Подходят ли стандартные криптографические протоколы?

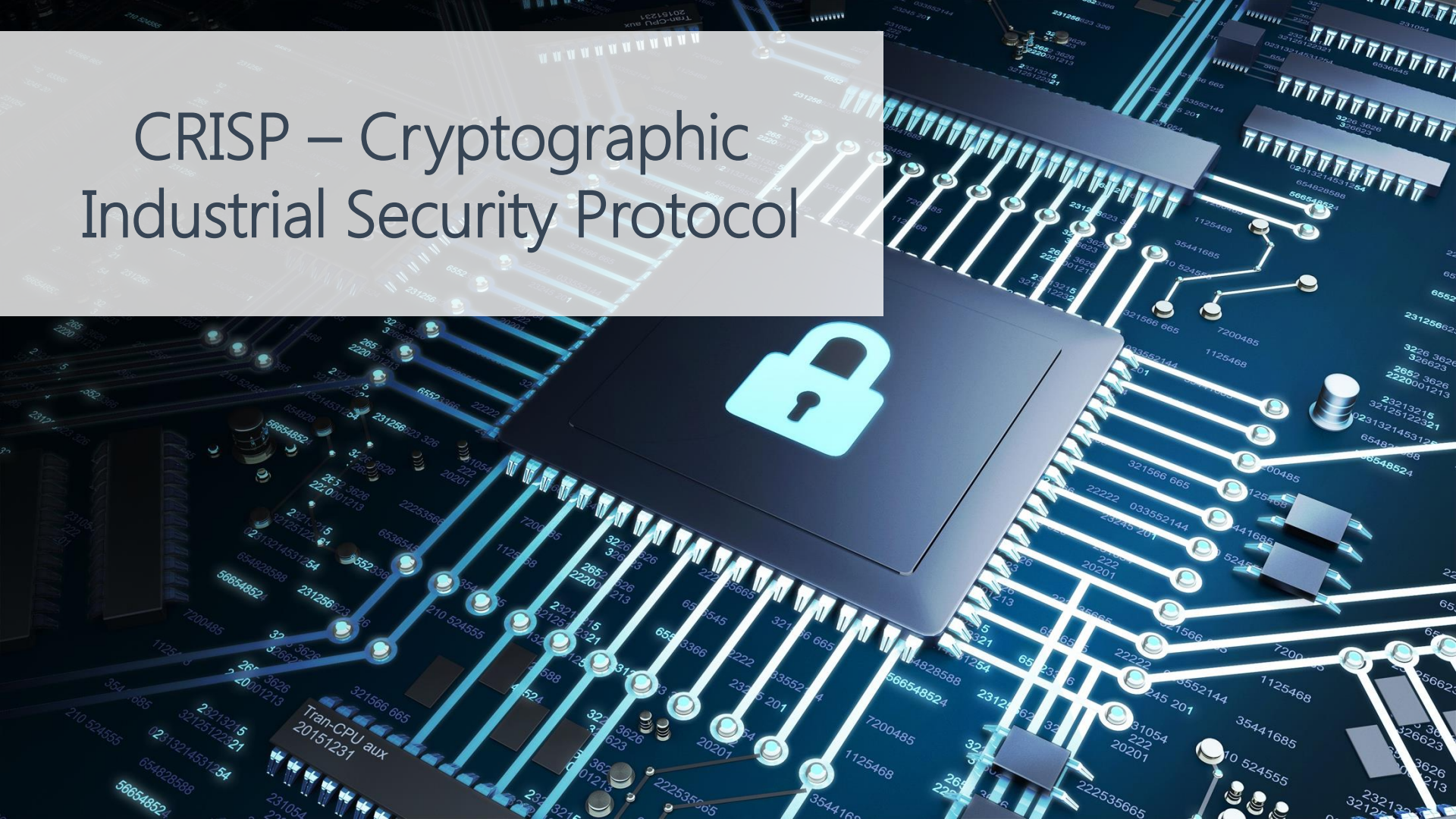
- IPSec – требует установления сессии, плохо работает на слабых каналах, IP based
- TLS – требует установления сессии, TCP/IP based
- CMS – большой оверхэд, большие задержки

Вывод: Рекомендованные в РФ криптографические протоколы не полностью решают поставленную задачу

Алгоритмы, основанные на PKI:

- Удобны для распределенных систем со сложной или неизвестной топологией
- Ресурсоёмкие (в том числе проблема энергопотребления)
- Медленные
- Часто не приспособлены для групповых коммуникаций

CRISP – Cryptographic Industrial Security Protocol



Криптографический протокол для M2M и IoT

Cryptographic Industrial Security Protocol

CRISP

Не всегда надежные каналы/ ограниченная пропускная способность
- без установления сессии -> предварительно распределенные ключи
- каждое сообщение несет всю необходимую информацию для обработки
- возможность приема сообщений не порядку

Целостность и аутентичность важнее конфиденциальности
- обязательная имитозащита
- опциональное шифрование

Минимальный overhead
- адресация абонентов может быть неявная, через протоколы целевой системы
- все криптографические детали определяются номером криптографического набора
- в качестве синхропосылки используется номер сообщения

Минимальные задержки обработки
- только симметричные механизмы
- минимальный набор механизмов
- один производный ключ может использоваться для обработки 2^{13} сообщений

Криптографический протокол CRISP

CRISP (Cryptographic Industrial Security Protocol) - неинтерактивный протокол защищенной передачи данных для индустриальных систем, M2M и IoT/IIoT коммуникаций

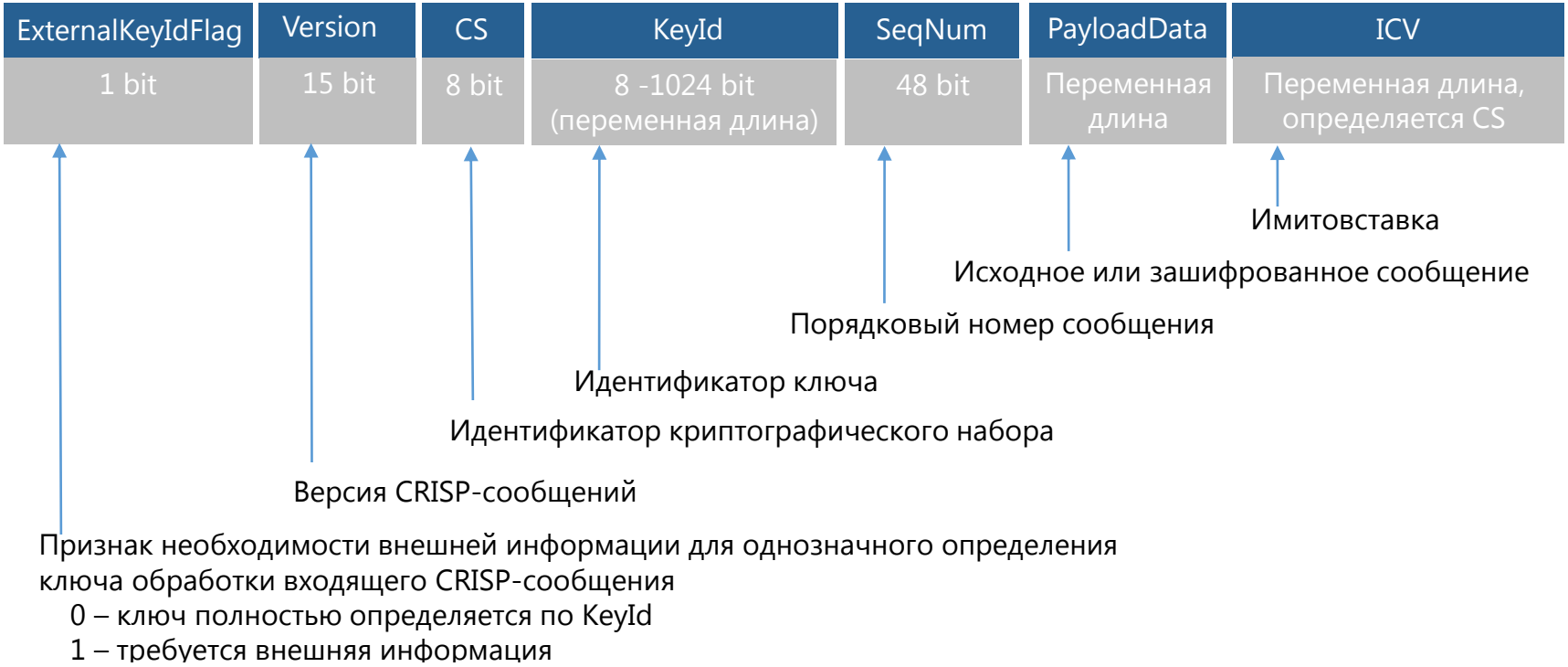
- Обеспечение имитозащиты
- Обеспечение конфиденциальности
- Защита от навязывания повторных сообщений
- Окно принятых сообщений

- У абонентов общий секретный ключ (может быть получен с помощью протоколов, основанных на PKI)
- Защита данных – блочный шифр, имитовставка
- Малый размер вспомогательных данных
- Поддержка адресных (один к одному) сообщений
- Поддержка многоадресных (один ко многим) сообщений
- Поддержка явной и неявной адресации абонентов

Криптографический протокол CRISP

- Не предназначен для встраивания в какой-либо определённый протокол передачи данных
- Представляет собой совокупность набора полей, правил их формирования и обработки
- На защищаемую систему возлагается задача доставки сформированных данных посредством используемых протоколов. В частности, адресация и маршрутизация данных возлагается на защищаемую систему

Структура CRISP-сообщений



Суммарный оверхэд – от **14 байтов**

Максимальный размер CRISP-сообщения – **2048 байтов**

CRISP: Защита исходящего сообщения

- Формируется номер сообщения $SeqNum$ – текущее значение $SeqNum$ увеличивается на 1
- Из базового ключа вырабатываются ключи шифрования (в случае, если криптографическим набором предусмотрено шифрование) и имитозащиты
- Формируются поля заголовка CRISP-сообщения – поля 1-5 таблицы 1
- Если криптографическим набором предусмотрено шифрование, то зашифровывается исходное сообщение
- Вычисляется значение имитовставки ICV для заголовка CRISP-сообщения и исходного сообщения (в случае, если криптографическим набором не предусмотрено шифрование) или заголовка CRISP-сообщения и зашифрованного сообщения (в случае, если криптографическим набором предусмотрено шифрование) – поля 1-6 таблицы 1

CRISP: Обработка входящего сообщения

- Если версия протокола CRISP, указанная в заголовке CRISP-сообщения, не поддерживается получателем, то сообщение блокируется
- Согласно значению $KeyId$ и, в случае $ExternalKeyIdFlag = 1$, дополнительной информации определяется базовый ключ
- Проверяется допустимость $SeqNum$ принятого CRISP-сообщения:
  - если $SeqNum$ принятого CRISP-сообщения меньше минимального номера окна принятых сообщений, то сообщение блокируется
  - если $SeqNum$ принятого CRISP-сообщения находится в пределах окна принятых сообщений и CRISP-сообщение с таким номером помечено как принятое, то сообщение блокируется
- Из базового ключа вырабатываются ключи шифрования (если криптографическим набором предусмотрено шифрование) и имитозащиты
- Выполняется контроль целостности CRISP-сообщения путём проверки имитовставки. Если имитовставка не верна, то сообщение блокируется

CRISP: Обработка входящего сообщения (2)

- Обновляется окно принятых сообщений:
  - если $SeqNum$ принятого CRISP-сообщения находится в пределах окна, то сообщение с таким номером помечается как принятое
  - если $SeqNum$ принятого CRISP-сообщения больше максимального номера окна принятых сообщений, то он становится новым максимальным номером, CRISP-сообщение с таким номером помечается как принятое и всё окно «сдвигается» в сторону нового максимального значения. То есть окно содержит список последовательных порядковых номеров входящих CRISP-сообщений, в котором помечены номера принятых CRISP-сообщений; максимальным номером окна становится $SeqNum$, а минимальным номером окна становится $SeqNum - Size + 1$ или 0, если $SeqNum - Size + 1 < 0$
- Если криптографическим набором предусмотрено шифрование, то выполняется расшифрование значения поля **PayloadData**, восстанавливается исходное сообщение.

CRISP: Идентификатор ключа KeyId

Заполнение поля KeyId

- 1000 0000 поле KeyId не используется
- 0xxx xxxx поле KeyId – 1 байт

Значение идентификатора

Значение идентификатора (от 1 до 127 байтов)

- 1yyy yyyy *zzzzz …. zzz* поле KeyId – y байт

Количество байтов в идентификатора

Использование поля KeyId

- Адреса участников (unicast, multicast)
- Различать ключи в случае сложных ключевых систем

CRISP: Механизмы защиты

Криптонабор CS=1

Целостность и аутентичность
- блочный шифр «Магма» в режиме выработки имитовставки по ГОСТ 34.13-2015

Конфиденциальность
- блочный шифр «Магма» в режиме гаммирования по ГОСТ 34.13-2015

Диверсификация ключей
- блочный шифр «Магма» в режиме выработки имитовставки
- контроль нагрузки на ключ/данные для диверсификации / синхропосылка – *SequenceNumber*

Защита от навязывания повторных сообщений
- счетчик сообщений *SequenceNumber* + скользящее окно принятых сообщений
- уникальность значений счетчика в срок действия одного базового ключа

Криптонабор CS=2

Целостность и аутентичность
- блочный шифр «Магма» в режиме выработки имитовставки по ГОСТ 34.13-2015

Диверсификация ключей
- блочный шифр «Магма» в режиме выработки имитовставки
- контроль нагрузки на ключ/данные для диверсификации – *SequenceNumber*

Защита от навязывания повторных сообщений
- счетчик сообщений *SequenceNumber* + скользящее окно принятых сообщений
- уникальность значений счетчика в срок действия одного базового ключа

CRISP: Диверсификация ключей

K_i = CMAC(*Key*,byte(*i*,1) || *Label* || *aL* || *SN* || *Node* || *CS* || *cL* || *oL*)

$K_{mac} = K_1 || K_2 || K_3 || K_4$ (*CS* = 1, *CS* = 2)
$K_{enc} = K_5 || K_6 || K_7 || K_8$ (*CS* = 1)

- *Key* - базовым ключом *K*
- *Label* = binary('macenc',6) (CS = 1)
  Label = binary('macmac',6) (CS = 2)
- *aL* = byte(6,1)
- *SN* = 0^5 || $\underline{MSB_{35}(SeqNum)}$
- *Node* - идентификатор отправителя;
- *CS* - значение поля **CS** CRISP-сообщения;
- *cL* = byte(*ContextLength*,2), где *ContextLength* равна сумме байтовых длин значений *SN*, *Node* и *CS*
- *OutputLength* = 512 (CS = 1)
  OutputLength = 256 (CS = 2)
- *oL* = byte(*OutputLength* ,2)

Планы по стандартизации

- Апрель 2019 – Утверждение текста методических рекомендация ТК26
- Сентябрь 2019 – Подготовка текста рекомендация по стандартизаций
- Декабрь 2019 – Утверждение рекомендаций по стандартизации

Перспективы развития

Разработка рекомендаций по использованию CRISP в протоколах M2M и IoT:

- ГОСТ Р МЭК 60870-5-104
- ГОСТ Р МЭК 60870-5-101
- Modbus TCP
- Modbus RTU
- GOOSE
- SV
- MQTT


infotecs
Спасибо за внимание!