ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 71252— 2024

Информационная технология

КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

Протокол защищенного обмена для индустриальных систем

Издание официальное

Москва
Российский институт стандартизации
2024

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Акционерным обществом «Информационные технологии и коммуникационные системы» (АО «ИнфоТеКС»)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 26 «Криптографическая защита информации»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 15 февраля 2024 г. № 235-ст

4 ВЗАМЕН Р 1323565.1.029—2019

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© Оформление. ФГБУ «Институт стандартизации», 2024

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения 1
4 Обозначения 2
5 Структура CRISP-сообщения 3
5.1 Перечень полей CRISP-сообщения 3
5.2 ExternalKeyIdFlag 3
5.3 Version 3
5.4 CS 3
5.5 KeyId 4
5.6 SeqNum 4
5.7 PayloadData 4
5.8 ICV 4
6 Ограничения 4
7 Обработка CRISP-сообщения 5
7.1 Инициализация порядкового номера сообщения и окна принятых сообщений 5
7.2 Защита исходного сообщения отправителем 5
7.3 Восстановление исходного сообщения получателем 5
8 Криптографические наборы 6
8.1 Набор MAGMA-CTR-CMAC 6
8.2 Набор MAGMA-NULL-CMAC 7
8.3 Набор MAGMA-CTR-CMAC8 8
8.4 Набор MAGMA-NULL-CMAC8 10
Приложение А (справочное) Контрольные примеры 12

Введение

Настоящий стандарт содержит описание протокола CRISP – CRyptographic Industrial Security Protocol – неинтерактивного протокола защищенной передачи данных, разработанного для применения в индустриальных системах. Протокол CRISP может быть использован для обеспечения конфиденциальности и аутентификации сообщений и для защиты от навязывания повторных сообщений.

Протокол CRISP реализует защиту исходных сообщений путем их опционального шифрования, а также вычисления имитовставки, в частности, для аутентификации сообщений и для защиты от навязывания повторных сообщений с использованием криптографических методов.

Протокол CRISP представляет собой совокупность набора полей, правил их формирования и обработки и может использоваться с любым протоколом передачи данных, способным доставить сформированные данные адресатам. При этом на защищаемую систему возлагается задача доставки сформированных данных посредством используемых протоколов. В частности, адресация и маршрутизация данных возлагается на защищаемую систему.

Примечание — Настоящий стандарт дополнен приложением А.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

Протокол защищенного обмена для индустриальных систем

Information technology. Cryptographic data security. Cryptographic industrial security protocol

Дата введения — 2024—04—01

1 Область применения

Настоящий стандарт описывает протокол CRISP, который применяется в системах с жесткими ограничениями на длину передаваемых данных, требующих использования неинтерактивных протоколов.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 34.12—2015 Информационная технология. Криптографическая защита информации. Блочные шифры

ГОСТ Р 34.13—2015 Информационная технология. Криптографическая защита информации. Режимы работы блочных шифров

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, применяется в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.1 **CRISP-сообщение:** Сообщение, защищенное с помощью протокола CRISP.

3.2 **базовый ключ:** Секретный ключ, известный только отправителю и получателю.

3.3 **идентификатор ключа:** Информация, использующаяся при определении ключа обработки CRISP-сообщения.

3.4 **индустриальная система:** Комплекс средств, обеспечивающий полный цикл функционирования производства или отдельного технологического процесса в различных областях экономики.

Издание официальное

3.5 **имитовставка:** Строка бит фиксированной длины, полученная применением симметричного криптографического метода к сообщению, добавляемая к сообщению для обеспечения его целостности и аутентификации источника данных.

3.6 **исходное сообщение:** Сообщение до защиты его протоколом CRISP.

3.7 **криптографический набор;** криптонабор: Совокупность криптографических алгоритмов и параметров, используемых в протоколе CRISP.

3.8 **неинтерактивный протокол:** Протокол, не требующий взаимного обмена сообщениями.

3.9 **окно принятых сообщений:** Диапазон допустимых порядковых номеров CRISP-сообщений, в котором помечены порядковые номера принятых CRISP-сообщений.

Примечание — Максимальным номером окна принятых сообщений является максимальный номер среди принятых CRISP-сообщений; минимальный номер окна принятых сообщений определяется максимальным номером окна принятых сообщений и размером окна принятых сообщений.

3.10 **отправитель:** Сторона, создающая CRISP-сообщение из исходного сообщения.

3.11 **получатель:** Сторона, восстанавливающая исходное сообщение из CRISP-сообщения.

3.12 **производный ключ:** Ключ шифрования сообщения или ключ вычисления имитовставки.

4 Обозначения

В настоящем стандарте использованы следующие обозначения:

V^* — множество всех двоичных строк конечной длины, включая пустую строку;

$|x|$ — длина (число компонент) строки $x \in V^*$;

V_s — множество всех двоичных строк длины s, где s — целое неотрицательное число; нумерация подстрок и компонент строки осуществляется справа налево, начиная с нуля;

$x||y$ — конкатенация двоичных строк x и y из V^*, т. е. строка из $V_{|x|+|y|}$, в которой подстрока с большими номерами компонент из $V_{|x|}$ совпадает со строкой x, а подстрока с меньшими номерами компонент из $V_{|y|}$ совпадает со строкой y;

V^l_8 — множество всех байтовых строк длины l, $l \geq 1$; имеет место соответствие между элементами множеств V^l_8 и V_{8l}, задаваемое равенством $(a_{l-1},\ldots,a_1,a_0) = x_{8l-1}||\ldots||x_1||x_0$, где $a_{l-1} = x_{8l-1}||\ldots||x_{8l-7}||x_{8l-8},\ldots, a_0 = x_7||\ldots||x_1||x_0$, $x_i \in V_1$, $i = 0,1,\ldots,8l-1$;

0^r — двоичная строка, состоящая из r нулей;

$\mathrm{LSB}_s(x): V^* \setminus \bigcup_{i=0}^{s-1} V_i \to V_s$ — отображение, ставящее в соответствие строке $z_{m-1}||\ldots||z_1||z_0$, $m \geq s$, строку $z_{s-1}||\ldots||z_1||z_0$, $z_i \in V_1$, $i = 0,1,\ldots,m-1$, $s \geq 1$;

$\mathrm{MSB}_s(x): V^* \setminus \bigcup_{i=0}^{s-1} V_i \to V_s$ — отображение, ставящее в соответствие строке $z_{m-1}||\ldots||z_1||z_0$, $m \geq s$, строку $z_{m-1}||\ldots||z_{m-s+1}||z_{m-s}$, $z_i \in V_1$, $i = 0,1,\ldots,m-1$, $s \geq 1$;

binary('*string*',*l*) — представление символьной строки *string*, состоящей из m символов, $m \geq 1$, в виде байтовой строки длины l, $l \geq m$, при котором сначала осуществляется посимвольный (с сохранением порядка следования символов) перевод исходной строки в байтовую строку $(a_{m-1},\ldots,a_1,a_0)$ длины m в соответствии с ASCII-представлением каждого символа, после чего в случае $l = m$ в качестве результата выдается байтовая строка $(a_{m-1},\ldots,a_1,a_0)$, а в случае $l > m$ — байтовая строка (0x00,…, 0x00, $a_{m-1},\ldots,a_1,a_0$) длины l;

byte(X,l) — представление целого числа X, $0 \leq X \leq 2^{8l-1}$, в виде байтовой строки длины l, $l \geq 1$, при котором соответствующая итоговой байтовой строке двоичная строка $x_{8l-1}||\ldots||x_1||x_0$, $x_i \in V_1$, $i = 0,1,\ldots, 8l-1$ есть бинарное представление числа X, т. е. $X = x_0 + x_1 \cdot 2 + \ldots + x_{8l-1} \cdot 2^{8l-1}$;

K_{ENC} — ключ шифрования сообщения;

K_{MAC} — ключ вычисления имитовставки;

K — базовый ключ;

Size — размер окна принятых сообщений.

5 Структура CRISP-сообщения

5.1 Перечень полей CRISP-сообщения

Здесь и далее названия полей сообщений выделяют прямым полужирным шрифтом. При указании конкретного значения поля используют курсив.

Для записи чисел используется сетевой порядок байтов (Big-endian).

Перечень полей CRISP-сообщения приведен в таблице 1.

Таблица 1 — Перечень полей CRISP-сообщения

Номер поля	Наименование поля		Длина поля в битах
1	Заголовок	**ExternalKeyIdFlag**	1
2		**Version**	15
3		**CS**	8
4		**KeyId**	От 8 до 1024
5		**SeqNum**	48
6	**PayloadData**		Переменная
7	**ICV**		Переменная, определяется значением **CS**

5.2 ExternalKeyIdFlag

Признак необходимости внешней информации для однозначного определения базового ключа для обработки входящего CRISP-сообщения. Длина поля — 1 бит.

ExternalKeyIdFlag = 0 означает, что базовый ключ для обработки входящего CRISP-сообщения однозначно определяется значением *KeyId. ExternalKeyIdFlag* = 1 означает, что для однозначного определения базового ключа для обработки входящего CRISP-сообщения требуется дополнительная информация.

5.3 Version

Версия протокола CRISP. Беззнаковое целое число. Длина поля — 15 бит.

Текущий документ описывает протокол CRISP, для которого *Version* = 0.

5.4 CS

Идентификатор криптографического набора. Беззнаковое целое число. Длина поля — 8 бит.

Идентификатор определяет криптографический набор, используемый для создания CRISP-сообщения или восстановления исходного сообщения из CRISP-сообщения. Всего может использоваться не более 256 различных криптонаборов, исходя из 8-битной длины поля CS CRISP-сообщения.

Список механизмов и параметров, определяемых и/или описываемых в криптографическом наборе, приведен в таблице 2.

Таблица 2 — Состав криптографического набора

Параметр	Описание	Правила задания	Назначение
EncryptionAlg	Алгоритм шифрования данных	Описание блочного шифра (или ссылка на такое описание); описание режима работы блочного шифра (или ссылка на такое описание), включая задание всех необходимых параметров	Алгоритм используется при шифровании сообщения (поля PayloadData)

Окончание таблицы 2

Параметр	Описание	Правила задания	Назначение
MACAlg	Алгоритм выработки имитовставки	Описание алгоритма (или ссылка на такое описание), включая задание всех необходимых параметров	Алгоритм используется при выработке имитовставки для полей 1–6 CRISP-сообщения
MACLength	Длина имитовставки	Длина имитовставки задается в байтах	—
DeriveIV	Алгоритм формирования синхропосылки	Описание алгоритма (или ссылки на такое описание); алгоритм должен быть согласован со спецификациями шифров и режимами их работы	Алгоритм используется для формирования синхропосылки при шифровании сообщения
DeriveKey	Алгоритмы выработки производных ключей из базового ключа	Описание алгоритмов, включая задание всех необходимых параметров	Алгоритмы используются для формирования ключей шифрования сообщения и ключей вычисления имитовставки

5.5 KeyId

Идентификатор ключа. Двоичная строка.

KeyId = 10000000_2 означает, что поле **KeyId** не используется. В остальных случаях:

- если $MSB_1(KeyId) = 0$, то длина поля **KeyId** составляет 1 байт и оставшиеся 7 бит содержат значение идентификатора ключа;
- если $MSB_1(KeyId) = 1$, то оставшиеся 7 бит первого байта интерпретируются как беззнаковое целое число и определяют количество дополнительных байтов (от 1 до 127). Дополнительные байты содержат значение идентификатора ключа.

5.6 SeqNum

Порядковый номер сообщения. Беззнаковое целое число. Длина поля — 48 бит.

Используется также для формирования синхропосылки алгоритма шифрования.

5.7 PayloadData

Исходное сообщение или зашифрованное исходное сообщение. Поле переменной длины.

Применение шифрования при обработке сообщения определяется использованным криптографическим набором (значением поля **CS**).

5.8 ICV

Имитовставка. Двоичная строка. Длина поля определяется использованным криптографическим набором (значением поля **CS**). Для конкретного значения CS длина поля **ICV** *м*ожет принимать только одно фиксированное значение.

Поле содержит значение имитовставки, рассчитанной для полей 1—6 CRISP-сообщения.

6 Ограничения

Максимальный размер CRISP-сообщения (суммарная длина всех полей CRISP-сообщения) — 2048 байт.

Предполагается следующее:

- отправитель и получатель имеют общий базовый ключ;
- с каждым базовым ключом ассоциирован идентификатор ключа, который может быть использован при определении базового ключа для обработки CRISP-сообщения;
- с каждым базовым ключом ассоциировано множество отправителей. Каждый из них обладает идентификатором отправителя *SourceIdentifier*, который может быть как внешней по отношению к *KeyId*

информацией, так и частью *KeyId*. Идентификатор однозначно определяет отправителя для каждого базового ключа, т. е. у разных отправителей, имеющих общий базовый ключ, *SourceIdentifier* различны;

- отправитель и получатель имеют общие криптографические наборы;
- задача определения базового ключа обработки сообщения отправителем и получателем находится за рамками протокола CRISP;
- для каждого отправителя и получателя настроен размер окна принятых сообщений.

Примечания

1 Способ установки на отправителе и получателе общего базового ключа, его идентификатора, криптографических наборов и *SourceIdentifier* находится за рамками протокола CRISP.

2 Под определением базового ключа для обработки сообщения отправителем или получателем понимается поиск нужного ключа среди установленных на отправителе или получателе на основании *KeyId* и, при необходимости, дополнительной информации, например *SourceIdentifier*.

3 Размер идентификатора отправителя *SourceIdentifier* должен быть в пределах от 4 до 32 байт.

4 Размер окна принятых сообщений *Size* должен быть в пределах от 1 до 256 (включительно) сообщений.

7 Обработка CRISP-сообщения

7.1 Инициализация порядкового номера сообщения и окна принятых сообщений

Перед первым использованием конкретного базового ключа с целью защиты сообщений отправитель устанавливает начальное значение (инициализирует) *SeqNum*. Инициализация *SeqNum* может потребоваться также для восстановления после сбоев в нумерации сообщений.

Настоящий стандарт не специфицирует конкретный алгоритм инициализации *SeqNum*. При этом алгоритм инициализации совместно с правилом формирования порядкового номера исходящих сообщений должны обеспечивать нахождение значения *SeqNum* в диапазоне от 0 до 2^{48}–1 (включительно) и строгое возрастание значения *SeqNum* для каждого сообщения, создаваемого одним отправителем с использованием одного базового ключа.

Перед первым использованием конкретного базового ключа для каждого ассоциированного с ним отправителя с целью восстановления сообщений получатель устанавливает максимальный и минимальный номера окна принятых сообщений равными 0.

7.2 Защита исходного сообщения отправителем

Предполагается, что определен базовый ключ, его идентификатор и криптонабор для формирования CRISP-сообщения данному получателю.

Для создания CRISP-сообщения выполняется такая последовательность действий:

- формируется порядковый номер сообщения *SeqNum* — текущее значение *SeqNum* увеличивается на 1;
- из базового ключа вырабатываются ключи шифрования (в случае, если криптографическим набором предусмотрено шифрование) и вычисления имитовставки;
- формируются поля заголовка CRISP-сообщения — поля 1—5 таблицы 1;
- если криптографическим набором предусмотрено шифрование, то зашифровывается исходное сообщение;
- вычисляется значение имитовставки *ICV* от содержимого полей 1—6 таблицы 1, т. е. 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).

7.3 Восстановление исходного сообщения получателем

Для восстановления исходного сообщения из CRISP-сообщения выполняется такая последовательность действий:

а) если версия протокола CRISP *Version* или идентификатор криптографического набора *CS*, указанные в заголовке, не поддерживаются получателем, то обработка CRISP-сообщения прекращается;

б) согласно значению *KeyId* и в случае *ExternalKeyIdFlag* = 1 дополнительной информации определяется базовый ключ, устанавливается отправитель и выбирается окно принятых сообщений от данного отправителя;

в) проверяется допустимость значения *SeqNum* входящего CRISP-сообщения:

1) если значение *SeqNum* входящего CRISP-сообщения меньше минимального номера окна принятых сообщений, то обработка CRISP-сообщения прекращается;

2) если значение *SeqNum* входящего CRISP-сообщения принадлежит окну принятых сообщений и данный порядковый номер CRISP-сообщения помечен как принятый в окне принятых сообщений, то обработка CRISP-сообщения прекращается;

3) в остальных случаях значение *SeqNum* входящего CRISP-сообщения является допустимым и осуществляется переход к следующему шагу;

г) из базового ключа вырабатываются ключи шифрования (если криптографическим набором предусмотрено шифрование) и вычисления имитовставки;

д) выполняется контроль целостности CRISP-сообщения путем проверки имитовставки:

1) если имитовставка, рассчитанная для полей 1—6 принятого CRISP-сообщения, неверна (не совпадает со значением поля **ICV**), то обработка CRISP-сообщения прекращается;

2) если имитовставка верна (совпадает со значением поля **ICV**), то осуществляется переход к следующему шагу;

е) обновляется окно принятых сообщений:

1) если значение *SeqNum* входящего CRISP-сообщения принадлежит окну принятых сообщений, то порядковый номер *SeqNum* помечается в окне принятых сообщений как принятый;

2) если значение *SeqNum* входящего CRISP-сообщения больше максимального номера окна принятых сообщений, то новым максимальным номером окна принятых сообщений становится значение *SeqNum*, порядковый номер *SeqNum* помечается в окне принятых сообщений как принятый, а новым минимальным номером окна принятых сообщений становится значение *SeqNum* – *Size* + 1 или 0, если значение *SeqNum* − *Size* + 1 меньше 0;

ж) извлекается исходное сообщение из поля **PayloadData** и, если криптографическим набором предусмотрено шифрование, то выполняется его расшифрование.

8 Криптографические наборы

8.1 Набор MAGMA-CTR-CMAC

8.1.1 Описание криптографического набора MAGMA-CTR-CMAC

Значение идентификатора данного криптонабора: *CS* = 1.

Описание криптографического набора MAGMA-CTR-CMAC приведено в таблице 3.

Таблица 3 — Описание криптографического набора MAGMA-CTR-CMAC

Параметр	Значение
EncryptionAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме гаммирования согласно ГОСТ Р 34.13—2015 (пункт 5.2)
MACAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6)
MACLength	4 байта
DeriveIV	Описание приведено в 8.1.4
DeriveKey	Описание приведено в 8.1.5

8.1.2 Алгоритм шифрования

Для шифрования данных используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме гаммирования согласно ГОСТ Р 34.13—2015 (пункт 5.2). В качестве ключа используется ключ шифрования сообщения K_{ENC}. Значение входного параметра режима гаммирования $s = 64$. В качестве синхропосылки используется значение, определенное в 8.1.4. Дополнение сообщений для режима гаммирования не предусмотрено.

8.1.3 Алгоритм выработки имитовставки

Для вычисления имитовставки *ICV*, содержащейся в поле **ICV** CRISP-сообщения, используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6). В качестве ключа используется ключ вычисления имитовставки K_{MAC}. Значение входного параметра режима выработки имитовставки s = 32. Криптографическое дополнение данных для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

8.1.4 Алгоритм формирования синхропосылки

Для формирования из 48-битного порядкового номера сообщения *SeqNum,* содержащегося в поле **SeqNum** CRISP-сообщения, 32-битной синхропосылки *IV* используются 32 младших бита *SeqNum*:

$$IV = \mathrm{LSB}_{32}(\mathrm{byte}(SeqNum, 6)).$$

8.1.5 Алгоритм выработки производных ключей

Для выработки ключей шифрования сообщения и вычисления имитовставки используется функция

$$\mathrm{CMAC}(Key,Data),$$

которая реализуется с помощью блочного шифра «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6) для данных *Data* на ключе *Key*. Значение входного параметра режима выработки имитовставки s = 64. Криптографическое дополнение для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

Для выработки ключа шифрования сообщения и ключа вычисления имитовставки используется следующий алгоритм:

$$K_{MAC} = K_1 \,||\, K_2 \,||\, K_3 \,||\, K_4;$$

$$K_{ENC} = K_5 \,||\, K_6 \,||\, K_7 \,||\, K_8.$$

Для каждого числа i = 1,…,8 вычисляются 64-битные величины:

$$K_i = \mathrm{CMAC}(Key,\mathrm{byte}(i, 1) \,||\, Label \,||\, aL \,||\, SN \,||\, Node \,||\, \mathrm{byte}(CS,1) \,||\, cL \,||\, oL),$$

где:

- *Key* — инициализируется базовым ключом *K*;
- *Label* = binary(‘macenc’, 6);
- *aL* = byte(6,1);
- $SN = 0^5 \,||\, \mathrm{MSB}_{35}(\mathrm{byte}(SeqNum, 6))$, где *SeqNum* инициализируется значением *SeqNum* CRISP-сообщения;
- *Node* = *SourceIdentifier*, где *SourceIdentifier* инициализируется значением идентификатора отправителя;
- *CS* — инициализируется значением *CS* CRISP-сообщения;
- *cL* = byte(*ContextLength*, 2), где *ContextLength* — сумма длин (в байтах) значений *SN*, *Node* и *CS*;
- *OutputLength* = 512, где *OutputLength* — необходимая битовая длина вырабатываемого ключевого материала;
- *oL* = byte(*OutputLength*, 2).

8.2 Набор MAGMA-NULL-CMAC

8.2.1 Описание криптографического набора MAGMA-NULL-CMAC

Значение идентификатора данного криптонабора: *CS* = 2.

Описание криптографического набора MAGMA-NULL-CMAC приведено в таблице 4.

Таблица 4 — Описание криптографического набора MAGMA-NULL-CMAC

Параметр	Значение
EncryptionAlg	Не используется
MACAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6)
MACLength	4 байта
DeriveIV	Не используется
DeriveKey	Описание приведено в 8.2.3

8.2.2 Алгоритм выработки имитовставки

Для вычисления имитовставки *ICV*, содержащейся в поле **ICV** CRISP-сообщения, используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6). В качестве ключа используется ключ вычисления имитовставки K_{MAC}. Значение входного параметра режима выработки имитовставки s = 32. Криптографическое дополнение данных для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

8.2.3 Алгоритм выработки производных ключей

Для выработки ключа вычисления имитовставки используется функция

$$\mathrm{CMAC}(Key, Data),$$

которая реализуется с помощью блочного шифра «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6) для данных *Data* на ключе *Key*. Значение входного параметра режима выработки имитовставки s = 64. Криптографическое дополнение для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

Для выработки ключа вычисления имитовставки используется следующий алгоритм:

$$K_{MAC} = K_1 \,||\, K_2 \,||\, K_3 \,||\, K_4.$$

Для каждого числа i = 1,…,4 вычисляются 64-битные величины:

$$K_i = \mathrm{CMAC}(Key, \mathrm{byte}(i,1) \,||\, Label \,||\, aL \,||\, SN \,||\, Node \,||\, \mathrm{byte}(CS,1) \,||\, cL \,||\, oL),$$

где:

- *Key* — инициализируется базовым ключом *K*;
- *Label* = binary(‘macmac’, 6);
- *aL* = byte(6,1);
- $SN = 0^5 \,||\, \mathrm{MSB}_{35}(\mathrm{byte}(SeqNum, 6))$, где *SeqNum* инициализируется значением *SeqNum* CRISP-сообщения;
- *Node* = *SourceIdentifier*, где *SourceIdentifier* инициализируется значением идентификатора отправителя;
- *CS* — инициализируется значением *CS* CRISP-сообщения;
- *cL* = byte(*ContextLength*, 2), где *ContextLength* — сумма длин (в байтах) значений *SN*, *Node* и *CS*;
- *OutputLength* = 256, где *OutputLength* — необходимая битовая длина вырабатываемого ключевого материала;
- *oL* = byte(*OutputLength*, 2).

8.3 Набор MAGMA-CTR-CMAC8

8.3.1 Описание криптографического набора MAGMA-CTR-CMAC8

Значение идентификатора данного криптонабора: *CS* = 3.

Описание криптографического набора MAGMA-CTR-CMAC8 приведено в таблице 5.

Таблица 5 — Описание криптографического набора MAGMA-CTR-CMAC8

Параметр	Значение
EncryptionAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме гаммирования согласно ГОСТ Р 34.13—2015 (пункт 5.2)
MACAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6)
MACLength	8 байт
DeriveIV	Описание приведено в 8.3.4
DeriveKey	Описание приведено в 8.3.5

8.3.2 Алгоритм шифрования

Для шифрования данных используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме гаммирования согласно ГОСТ Р 34.13—2015 (пункт 5.2). В качестве ключа используется ключ шифрования сообщения K_{ENC}. Значение входного параметра режима гаммирования $s = 64$. В качестве синхропосылки используется значение, определенное в 8.3.4. Дополнение сообщений для режима гаммирования не предусмотрено.

8.3.3 Алгоритм выработки имитовставки

Для вычисления имитовставки *ICV*, содержащейся в поле **ICV** CRISP-сообщения, используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6). В качестве ключа используется ключ вычисления имитовставки K_{MAC}. Значение входного параметра режима выработки имитовставки $s = 64$. Криптографическое дополнение данных для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

8.3.4 Алгоритм формирования синхропосылки

Для формирования из 48-битного порядкового номера сообщения *SeqNum,* содержащегося в поле **SeqNum** CRISP-сообщения, 32-битной синхропосылки *IV* используются 32 младших бита *SeqNum*

$$IV = \mathrm{LSB}_{32}(\mathrm{byte}(SeqNum, 6)).$$

8.3.5 Алгоритм выработки производных ключей

Для выработки ключей шифрования сообщения и вычисления имитовставки используется функция

$$\mathrm{CMAC}(Key, Data),$$

которая реализуется с помощью блочного шифра «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6) для данных *Data* на ключе *Key*. Значение входного параметра режима выработки имитовставки $s = 64$. Криптографическое дополнение для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

Для выработки ключа шифрования сообщения и ключа вычисления имитовставки используется следующий алгоритм:

$$K_{MAC} = K_1 \,||\, K_2 \,||\, K_3 \,||\, K_4;$$

$$K_{ENC} = K_5 \,||\, K_6 \,||\, K_7 \,||\, K_8.$$

Для каждого числа $i = 1,\ldots,8$ вычисляются 64-битные величины:

$$K_i = \mathrm{CMAC}(Key, \mathrm{byte}(i,1) \,||\, Label \,||\, aL \,||\, SN \,||\, Node \,||\, \mathrm{byte}(CS,1) \,||\, cL \,||\, oL),$$

где:

- *Key* — инициализируется базовым ключом *K*;
- *Label* = binary(‘macenc’, 6);

- *aL* = byte(6, 1);
- *SN* = 0^5 || MSB_{35}(byte(*SeqNum*, 6)), где *SeqNum* инициализируется значением *SeqNum* CRISP-сообщения;
- *Node* = *SourceIdentifier*, где *SourceIdentifier* инициализируется значением идентификатора отправителя;
- *CS* — инициализируется значением *CS* CRISP-сообщения;
- *cL* = byte(*ContextLength*, 2), где *ContextLength* — сумма длин (в байтах) значений *SN*, *Node* и *CS*;
- *OutputLength* = 512, где *OutputLength* — необходимая битовая длина вырабатываемого ключевого материала;
- *oL* = byte(*OutputLength*, 2).

8.4 Набор MAGMA-NULL-CMAC8

8.4.1 Описание криптографического набора MAGMA-NULL-CMAC8

Значение идентификатора данного криптонабора: *CS* = 4.

Описание криптографического набора MAGMA-NULL-CMAC8 приведено в таблице 6.

Таблица 6 — Описание криптографического набора MAGMA-NULL-CMAC8

Параметр	Значение
EncryptionAlg	Не используется
MACAlg	Блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6)
MACLength	8 байт
DeriveIV	Не используется
DeriveKey	Описание приведено в 8.4.3

8.4.2 Алгоритм выработки имитовставки

Для вычисления имитовставки *ICV*, содержащейся в поле **ICV** CRISP-сообщения, используется блочный шифр «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6). В качестве ключа используется ключ вычисления имитовставки K_{MAC}. Значение входного параметра режима выработки имитовставки *s* = 64. Криптографическое дополнение данных для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

8.4.3 Алгоритм выработки производных ключей

Для выработки ключа вычисления имитовставки используется функция

$$\text{CMAC}(Key, Data),$$

которая реализуется с помощью блочного шифра «Магма» согласно ГОСТ Р 34.12—2015 (раздел 5) в режиме выработки имитовставки согласно ГОСТ Р 34.13—2015 (пункт 5.6) для данных *Data* на ключе *Key*. Значение входного параметра режима выработки имитовставки *s* = 64. Криптографическое дополнение для режима выработки имитовставки выполняется согласно ГОСТ Р 34.13—2015 (пункт 4.1).

Для выработки ключа вычисления имитовставки используется следующий алгоритм

$$K_{MAC} = K_1 \,||\, K_2 \,||\, K_3 \,||\, K_4.$$

Для каждого числа $i = 1,\ldots,4$ вычисляются 64-битные величины:

$$K_i = \text{CMAC}(Key, \text{byte}(i,1) \,||\, Label \,||\, aL \,||\, SN \,||\, Node \,||\, \text{byte}(CS,1) \,||\, cL \,||\, oL),$$

где:

- *Key* — инициализируется базовым ключом *K*;
- *Label* = binary('macmac', 6);
- *aL* = byte(6, 1);
- *SN* = 0^5 || MSB_{35}(byte(*SeqNum*, 6)), где *SeqNum* инициализируется значением *SeqNum* CRISP-сообщения;

- *Node* = *SourceIdentifier*, где *SourceIdentifier* инициализируется значением идентификатора отправителя;
- *CS* — инициализируется значением *CS* CRISP-сообщения;
- *cL* = byte(*ContextLength*,2), где *ContextLength* — сумма длин (в байтах) значений *SN*, *Node* и *CS*;
- *OutputLength* = 256, где *OutputLength* — необходимая битовая длина вырабатываемого ключевого материала;
- *oL* = byte(*OutputLength*, 2).

Приложение А
(справочное)

Контрольные примеры

Приводимые ниже значения *SourceIdentifier, KeyId, SeqNum* и базового ключа *K* рекомендуется использовать только для проверки корректной работы конкретной реализации алгоритмов, описанных в настоящем стандарте.

Все числовые значения приведены в шестнадцатеричной системе счисления.

В данном приложении двоичные строки из V^*, длина которых кратна 4, записываются в шестнадцатеричном виде, а символ конкатенации опускается. То есть строка $a \in V_{4r}$ будет представлена в виде $a_{r-1} - a_{r-2} \cdots a_0$, где $a_i \in \{0, 1, \ldots, 9, a, b, c, d, e, f\}$, $i = 0, 1, \ldots, r-1$. Соответствие между двоичными строками длины 4 и шестнадцатеричными строками длины 1 задается естественным образом (таблица А.1).

Преобразование, ставящее в соответствие двоичной строке длины $4r$ шестнадцатеричную строку длины r, и соответствующее обратное преобразование для простоты записи опускается.

Таблица А.1 — Соответствие между двоичными и шестнадцатеричными строками

Двоичная запись	Шестнадцатеричная запись
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	a
1011	b
1100	c
1101	d
1110	e
1111	f

Примечание — Символ «\\» обозначает перенос числа на новую строку.

А.1 Набор MAGMA-CTR-CMAC: CS = 1

Для формирования сообщения используются следующие значения:

ExternalKeyIdFlag = 1

Version = 0

CS = 01_{16}

KeyId = 30_{16}

SeqNum = $0b76e6736001_{16}$

SourceIdentifier = $303230353138303030303031_{16}$

K = 56509427153249653498524659324653\\
$04532945346593845073249576351290_{16}$

Исходное сообщение:

PayloadData = 486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03$_{16}$

На основе исходных данных получаются следующие значения ключа вычисления имитовставки K_{MAC}, ключа шифрования K_{ENC}, зашифрованного сообщения и имитовставки *ICV*:

K_{MAC} = eeb0f6814257ad08964eabe5e0993d38\\
b2afc2ada24e8362d455db06951f2d93$_{16}$

K_{ENC} = e3316ad28c788c38dafdeb9388e234bd\\
30e5c901eeeb1788cdc1ec5db315e1a7$_{16}$

ICV = 887f0a32$_{16}$

Зашифрованное сообщение:

d324643aefd97b93b18d343a2fba477e\\
c704cd8d14ac1cf74ceb25577af8fc2c\\
25fa9050a1$_{16}$

Итоговое сообщение будет иметь следующий вид:

800001300b76e6736001\\
d324643aefd97b93b18d343a2fba477e\\
c704cd8d14ac1cf74ceb25577af8fc2c\\
25fa9050a1\\
887f0a32$_{16}$

А.2 Набор MAGMA-NULL-CMAC: CS = 2

Для формирования сообщения используются следующие значения:

ExternalKeyIdFlag = 1

Version = 0

CS = 02$_{16}$

KeyId = 30$_{16}$

SeqNum = 0b76e66ea001$_{16}$

SourceIdentifier = 303230353138303030303031$_{16}$

K = 56509427153249653498524659324653\\
0453294534659384507324957635 1290$_{16}$

Исходное сообщение:

PayloadData = 486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03$_{16}$

На основе исходных данных получаются следующие значения ключа вычисления имитовставки K_{MAC} и имитовставки *ICV*:

K_{MAC} = c3e3780f87f2caf539fdad56d9cb0340\\
b1052c0ae8272ddc9601c921f81a7ca5$_{16}$

ICV = b97ade94$_{16}$

Итоговое сообщение будет иметь следующий вид:

800002300b76e66ea001\\
486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03\\
b97ade94$_{16}$

А.3 Набор MAGMA-CTR-CMAC8: CS = 3

Для формирования CRISP-сообщения используются следующие значения:

ExternalKeyIdFlag = 1

Version = 0

CS = 03$_{16}$

KeyId = 30$_{16}$

SeqNum = 0b76e6736001$_{16}$

SourceIdentifier = 303230353138303030303031$_{16}$

K = 56509427153249653498524659324653\\
0453294534659384507324957635 1290$_{16}$

Исходное сообщение:

PayloadData = 486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03$_{16}$

На основе исходных данных получаются следующие значения ключа вычисления имитовставки K_{MAC}, ключа шифрования K_{ENC}, зашифрованного сообщения и имитовставки *ICV*:

K_{MAC} = 742ae2acebae5fed1cc7acfd614d9cf2\\
98aeeea7a77a997bc19b99b9beeb8832$_{16}$

K_{ENC} = c241ebfac49d476859e1e6388a94660b\\
65d6b740a38363abb9129297250ddb22$_{16}$

ICV = edf339a0dbc0b5b7$_{16}$

Зашифрованное сообщение:

9def18d705afde4e00edb132a8b8d480\\
18ffe760fdd34cecd6461c3553c3087c\\
d0756f1569$_{16}$

Итоговое сообщение будет иметь следующий вид:

800003300b76e6736001\\
9def18d705afde4e00edb132a8b8d480\\
18ffe760fdd34cecd6461c3553c3087c\\
d0756f1569\\
edf339a0dbc0b5b7$_{16}$

A.4 Набор MAGMA-NULL-CMAC8: CS = 4

Для формирования сообщения используются следующие значения:

ExternalKeyIdFlag = 1

Version = 0

CS = 04$_{16}$

KeyId = 30$_{16}$

SeqNum = 0b76e66ea001$_{16}$

SourceIdentifier = 303230353138303030303031$_{16}$

K = 56509427153249653498524659324653\\
045329453465938450732495763512 90$_{16}$

Исходное сообщение:

PayloadData = 486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03$_{16}$

На основе исходных данных получаются следующие значения ключа вычисления имитовставки K_{MAC} и имитовставки *ICV*:

K_{MAC} = 5d4885a48e3bee6f79a3bd099ada4f68\\
21dcf81691d6710af3016c85ae06ebc4$_{16}$

ICV = f23152388e615825$_{16}$

Итоговое сообщение будет иметь следующий вид:

800004300b76e66ea001\\
486921205468697320697320746573 74\\
20666f72204352495350206d65737361\\
6765730a03\\
f23152388e615825$_{16}$

УДК 004.42:006.354 ОКС 35.040

Ключевые слова: криптографические протоколы, защищенная передача данных, индустриальные системы, аутентификация, шифрование, ключ

Редактор *Н.А. Аргунова*
Технический редактор *И.Е. Черепкова*
Корректор *О.В. Лазарева*
Компьютерная верстка *А.Н. Золотаревой*

Сдано в набор 19.02.2024. Подписано в печать 14.03.2024. Формат 60×84⅛. Гарнитура Ариал.
Усл. печ. л. 2,32. Уч.-изд. л. 1,90.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении в ФГБУ «Институт стандартизации» для комплектования Федерального информационного фонда стандартов, 117418 Москва, Нахимовский пр-т, д. 31, к. 2.

gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru