МЕЖДУНАРОДНЫЙ СОЮЗ ЭЛЕКТРОСВЯЗИ

МСЭ-Т

СЕКТОР СТАНДАРТИЗАЦИИ
ЭЛЕКТРОСВЯЗИ МСЭ

H.235.6

(09/2005)

СЕРИЯ H: АУДИОВИЗУАЛЬНЫЕ И МУЛЬТИМЕДИЙНЫЕ СИСТЕМЫ

Инфраструктура аудиовизуальных услуг – Системные аспекты

Средства защиты H.323: Профиль шифрования речевых сообщений с внутренним управлением ключами H.235/H.245

Рекомендация МСЭ-Т H.235.6

РЕКОМЕНДАЦИИ МСЭ-Т СЕРИИ H

АУДИОВИЗУАЛЬНЫЕ И МУЛЬТИМЕДИЙНЫЕ СИСТЕМЫ

ХАРАКТЕРИСТИКИ ВИДЕОТЕЛЕФОННЫХ СИСТЕМ	H.100–H.199
ИНФРАСТРУКТУРА АУДИОВИЗУАЛЬНЫХ УСЛУГ	
Общие положения	H.200–H.219
Мультиплексирование и синхронизация при передаче	H.220–H.229
Системные аспекты	**H.230–H.239**
Процедуры связи	H.240–H.259
Кодирование движущихся видеоизображений	H.260–H.279
Сопутствующие системные аспекты	H.280–H.299
Системы и оконечное оборудование для аудиовизуальных услуг	H.300–H.349
Архитектура услуг справочника для аудиовизуальных и мультимедийных услуг	H.350–H.359
Качество архитектуры обслуживания для аудиовизуальных и мультимедийных услуг	H.360–H.369
Дополнительные услуги для мультимедиа	H.450–H.499
ПРОЦЕДУРЫ МОБИЛЬНОСТИ И СОВМЕСТНОЙ РАБОТЫ	
Обзор мобильности и совместной работы, определений, протоколов и процедур	H.500–H.509
Мобильность для мультимедийных систем и услуг серии H	H.510–H.519
Приложения и услуги мобильной мультимедийной совместной работы	H.520–H.529
Защита мобильных мультимедийных систем и услуг	H.530–H.539
Защита приложений и услуг мобильной мультимедийной совместной работы	H.540–H.549
Процедуры мобильного взаимодействия	H.550–H.559
Процедуры взаимодействия мобильной мультимедийной совместной работы	H.560–H.569
ШИРОКОПОЛОСНЫЕ И МУЛЬТИМЕДИЙНЫЕ TRIPLE-PLAY УСЛУГИ	
Предоставление широкополосных мультимедийных услуг по VDSL	H.610–H.619

Для получения более подробной информации просьба обращаться к перечню Рекомендаций МСЭ-Т.

Рекомендация МСЭ-Т H.235.6

Средства защиты H.323: Профиль шифрования речевых сообщений с внутренним управлением ключами H.235/H.245

Резюме

Данная Рекомендация содержит процедуры защиты для профиля шифрования речевых сообщений (ранее в Приложении D/H.235), включая сопутствующее внутреннее управление ключами.

В предыдущих версиях подсерии H.235 данный профиль содержался в основной части H.235 и в ее Приложении D. В Дополнениях IV, V, VI к H.235.0 показано полное соответствие между пунктами, рисунками и таблицами версий 3 и 4 H.235.

Источник

Рекомендация МСЭ-Т H.235.6 утверждена 13 сентября 2005 года 16-й Исследовательской комиссией МСЭ-Т (2005–2008 гг.) в соответствии с процедурой, изложенной в Рекомендации МСЭ-Т A.8.

Ключевые слова

Аутентификация, сертификат, цифровая подпись, шифрование, целостность данных, управление ключами, мультимедийная защита, профиль защиты, шифрование речевых сообщений.

ПРЕДИСЛОВИЕ

Международный союз электросвязи (МСЭ) является специализированным учреждением Организации Объединенных Наций в области электросвязи. Сектор стандартизации электросвязи МСЭ (МСЭ-Т) – постоянный орган МСЭ. МСЭ-Т отвечает за изучение технических, эксплуатационных и тарифных вопросов и за выпуск Рекомендаций по ним с целью стандартизации электросвязи на всемирной основе.

На Всемирной ассамблее по стандартизации электросвязи (ВАСЭ), которая проводится каждые четыре года, определяются темы для изучения Исследовательскими комиссиями МСЭ-Т, которые, в свою очередь, вырабатывают Рекомендации по этим темам.

Утверждение Рекомендаций МСЭ-Т осуществляется в соответствии с процедурой, изложенной в Резолюции 1 ВАСЭ.

В некоторых областях информационных технологий, которые входят в компетенцию МСЭ-Т, необходимые стандарты разрабатываются на основе сотрудничества с ИСО и МЭК.

ПРИМЕЧАНИЕ

В настоящей Рекомендации термин "администрация" используется для краткости и обозначает как администрацию электросвязи, так и признанную эксплуатационную организацию.

Соблюдение положений данной Рекомендации носит добровольный характер. Однако в Рекомендации могут содержаться определенные обязательные положения (например, для обеспечения возможности взаимодействия или применимости), и соблюдение положений данной Рекомендации достигается в случае выполнения всех этих обязательных положений. Для выражения необходимости выполнения требований используется синтаксис долженствования и соответствующие слова (такие, как "должен" и т. п.), а также их отрицательные эквиваленты. Использование этих слов не предполагает, что соблюдение положений данной Рекомендации является обязательным для какой-либо из сторон.

ПРАВА ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

МСЭ обращает внимание на вероятность того, что практическое применение или реализация этой Рекомендации может включать использование заявленного права интеллектуальной собственности. МСЭ не занимает какую бы то ни было позицию относительно подтверждения, обоснованности или применимости заявленных прав интеллектуальной собственности, независимо от того, отстаиваются ли они членами МСЭ или другими сторонами вне процесса подготовки Рекомендации.

На момент утверждения настоящей Рекомендации МСЭ не получил извещение об интеллектуальной собственности, защищенной патентами, которые могут потребоваться для выполнения этой Рекомендации. Однако те, кто будет применять Рекомендацию, должны иметь в виду, что это может не отражать самую последнюю информацию, и поэтому им настоятельно рекомендуется обращаться к патентной базе данных БСЭ.

СОДЕРЖАНИЕ

Стр.

Рекомендация МСЭ-Т H.235.6

Средства защиты H.323: Профиль шифрования речевых сообщений с внутренним управлением ключами H.235/H.245

1 Сфера применения

Данная Рекомендация определяет профиль защиты для шифрования речевых сообщений, использующий внутреннее управление ключами H.235/H.245. В Рекомендации описаны процедуры шифрования речевых сообщений и соответствующего внутреннего управления ключами H.245.

2 Справочные документы

2.1 Нормативные справочные документы

В перечисленных ниже Рекомендациях МСЭ-Т и других справочных документах содержатся положения, которые, будучи упомянутыми в качестве справочных документов в данном тексте, составляют положения данной Рекомендации. На момент опубликования указанные издания были действующими. Все Рекомендации и другие справочные документы подлежат пересмотру, и поэтому всем пользователям этих Рекомендаций предлагается рассмотреть возможность использования самого последнего издания этих Рекомендаций и других справочных документов, перечисленных ниже. Перечень действующих Рекомендаций МСЭ-Т регулярно публикуется. Ссылка на любой документ в рамках данной Рекомендации не придает ему, даже притом, что это отдельный документ, статуса рекомендации.

– ITU-T Recommendation H.225.0 (2003), *Call signalling protocols and media stream packetization for packet-based multimedia communication systems*.

– ITU-T Recommendation H.235 version 1 (1998), *Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*.

– ITU-T Recommendation H.235 version 2 (2000), *Security and encryption for H series (H.323 and other H.245-based) multimedia terminals*.

– ITU-T Recommendation H.235 version 3 (2003), *Security and encryption for H series (H.323 and other H.245-based) multimedia terminals* plus Corrigendum 1 (2005).

– ITU-T Recommendation H.235.0 (2005), *H.323 security: Framework for security in H-series (H.323 and other H.245-based) multimedia systems*.

– ITU-T Recommendation H.235.1 (2005), *H.323 security: Baseline security profile*.

– ITU-T Recommendation H.235.2 (2005), *H.323 security: Signature security profile*.

– ITU-T Recommendation H.235.3 (2005), *H.323 security: Hybrid security profile*.

– ITU-T Recommendation H.245 (2005), *Control protocol for multimedia communication*.

– ITU-T Recommendation H.323 (2003), *Packet-based multimedia communications systems*.

– ITU-T Recommendation H.323 Annex F (1999), *Simple endpoint types*.

– ITU-T Recommendation X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications*.

– ISO 7498-2:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture*.

– ITU-T Recommendation X.803 (1994) | ISO/IEC 10745:1995, *Information technology – Open Systems Interconnection – Upper layers security model*.

– ITU-T Recommendation X.810 (1995) | ISO/IEC 10181-1:1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Overview*.

– ITU-T Recommendation X.811 (1995) | ISO/IEC 10181-2:1996, *Information technology – Open Systems Interconnection – Security frameworks for open systems: Authentication framework*.

– IETF RFC 2198 (1997), *RTP Payload for Redundant Audio Data*.

– IETF RFC 2246 (1999), *The TLS Protocol Version 1.0*.

– IETF RFC 2401 (1998), *Security Architecture for the Internet Protocol*.

– IETF RFC 2833 (2000), *RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*.

– IETF RFC 3546 (2003), *Transport Layer Security Protocol (TLS) Extensions*.

– US National Institute of Standards, "Advanced Encryption Algorithm (AES)", *Federal Information Processing Standard, (FIPS)* Publication 197, November 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

– ISO/IEC 9797-1:1999, *Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher*.

– ISO/IEC 9797-2:2002, *Information technology – Security techniques – Message Authentication Codes (MACs) – Part 2: Mechanisms using a dedicated hash-function*.

– ISO/IEC 10118-3:2004, *Information Technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions*.

– ISO/IEC 10116:2006, *Information technology – Security techniques – Modes of operation for an n-bit block cipher*.

2.2 Информативные справочные документы

[DES FIPS-46-2] US National Institute of Standards, Data Encryption Standard, *Federal Information Processing Standard*, (FIPS) Publication 46-2, December 1993, http://www.itl.nist.gov/fipspubs/fip46-2.htm.

[DES FIPS-74] US National Institute of Standards, Guidelines for Implementing and Using the Data Encryption Standard, *Federal Information Processing Standard*, (FIPS) Publication 74, April 1981, http://www.itl.nist.gov/div897/pubs/fip74.htm.

[DES FIPS-81] US National Institute of Standards, DES Modes of Operation, *Federal Information Processing Standard*, (FIPS) Publication 81, December 1980, http://www.itl.nist.gov/fipspubs/fip81.htm.

[FIPS PUB 180-1] NIST, FIPS PUB 180-1: *Secure Hash Standard*, April 1995 http://csrc.nist.gov/fips/fip180-1.ps.

[LI] ETSI TR 101 772 V1.1.2, Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 3; Service Independent requirements definition; Lawful interception – top level requirements.

[OIW] Stable Implementation – Agreements for Open Systems Interconnection Protocols: Part 12 – OS Security; Output from the December 1994 Open Systems Environment Implementors' Workshop (OIW); http://nemo.ncsl.nist.gov/oiw/agreements/stable/OSI/12s_9412.txt.

[RFC2268] IETF RFC 2268 (1998), *A Description of the RC2$^{(r)}$ Encryption Algorithm*.

[RFC2405] IETF RFC 2405 (1998), *The ESP DES-CBC Cipher Algorithm With Explicit IV*.

[RFC2412] IETF RFC 2412 (1998), *The OAKLEY Key Determination Protocol*.

[WEBOIDs] http://www.alvestrand.no/objectid/top.html.

[Daemon] DAEMON (J.), *Cipher and Hash function design*, Ph.D. Thesis, Katholieke Universiteit Leuven, March 1995.

[ESP] IETF RFC 2406 (1998), *IP Encapsulating Security Payload (ESP)*.

[IKE] IETF RFC 2409 (1998), *The Internet Key Exchange (IKE)*.

[ISAKMP] IETF RFC 2408 (1998), *Internet Security Association and Key Management Protocol (ISAKMP)*.

[J.170] ITU-T Recommendation J.170 (2005), *IPCablecom security specification*.

[RTP] IETF RFC 3550 (2003), *RTP: A transport Protocol for Real-Time Applications*.

[Schneier] SCHNEIER (B.), *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd Edition, John Wiley & Sons, Inc., 1995.

[SRTP] IETF RFC 3711 (2004), *The Secure Real-Time Transport Protocol*.

3 Термины и определения

Для целей данной Рекомендации применяются определения, данные в пунктах 3/H.323, 3/H.225.0 и 3/H.245. Некоторые из терминов, используемых в данной Рекомендации, также определяются в Рекомендациях МСЭ-Т X.800 | ИСО 7498-2, X.803 | ИСО/МЭК 10745, X.810 | ИСО/МЭК 10181-1 и X.811 | ИСО/МЭК 10181-2.

С одной стороны, сеансовый ключ для шифрования медиапотоков генерируется ведущим только для определенного сеанса RTP (на OLC), с максимальной длительностью в один вызов. Полученный сеансовый ключ шифруется ключом, производным от согласованного разделяемого секрета Диффи-Хеллмана, обе конечные точки которого рассчитаны. В этом случае DH-разделяемый секрет действует как главный ключ (Master-Key) для защиты сеансового(ых) ключа(ей).

4 Символы и аббревиатуры

В данной Рекомендации используются следующие сокращения:

3DES	Triple DES	Тройной DES	
AES	Advanced Encryption Algorithm	Усовершенствованный алгоритм шифрования	
ASN.1	Abstract Syntax Notation One	Абстрактно-синтаксическая нотация версии 1	
CBC	Cipher Block Chaining	Сцепление шифрованных блоков	
CFB	Cipher Feedback	Обратная связь по шифрованному тексту	
DES	Data Encryption Standard	Стандарт шифрования данных	
DH	Diffie-Hellman	Алгоритм Диффи-Хеллмана	
DTMF	Dual Tone Multiple Frequency	Двухтональный многочастотный	
ECB	Electronic Code Book	Электронная кодовая книга	
EOFB	Enhanced Output Feedback Mode	Усовершенствованный режим обратной связи по выходу	
EP	Endpoint	Конечная точка	
FEC	Forward Error Correction	Упреждающая коррекция ошибок	
GK	Gatekeeper	Контроллер доступа	
HMAC	Hashed Message Authentication Code	Код аутентификации сообщений с помощью хеш-функции	
IPsec	Internet Protocol Security	Защита на уровне протокола Интернет	
ITU	International Telecommunication Union	Международный союз электросвязи	МСЭ
IV	Initialization Vector	Вектор инициализации	
KS	Salting Key in EOFB mode	Расширенный ключ в режиме EOFB	
MAC	Message Authentication Code	Код аутентификации сообщений	

MC	Multipoint Controller	Контроллер многоточечной связи
MCU	Multipoint Control Unit	Блок управления многоточечной связью
MPS	Multiple Payload Stream	Поток групповой полезной нагрузки
OFB	Output Feedback Mode	Режим обратной связи по выходу
OID	Object Identifier	Идентификатор объекта
OLC	Open Logical Channel	Открытый логический канал
RAS	Registration, Admission and Status	Регистрация, допуск и статус
RC	Rivest Cipher	Шифр Райвеста
ROC	Roll-over Counter	Автоматически переводящийся счетчик
RSA	Rivest, Shamir, Adleman	Алгоритм шифрования Райвеста-Шамира-Адлемана
RTP	Real-Time Protocol	Протокол передачи данных в режиме реального времени
RTCP	Real-Time Control Protocol	Протокол управления передачей данных в режиме реального времени
SDU	Service Data Unit	Сервисный блок данных
SEQ	Sequence number	Номер последовательности
SHA	Secure Hash Algorithm	Алгоритм аутентификации и проверки целостности информации, алгоритм SHA
TCP	Transmission Control Protocol	Протокол управления передачей
TLS	Transport Layer Security	Защита транспортного уровня
TSAP	Transport Service Access Point	Точка доступа к услугам транспортного уровня
UDP	User Datagram Protocol	Протокол датаграмм пользователя
XOR	Exclusive OR	Исключающее ИЛИ

5 Соглашения

В данной Рекомендации используются следующие соглашения:

– "должен" указывает на обязательное требование,

– "следует" указывает на предполагающийся, но не обязательный образ действия,

– "может" означает скорее на необязательный образ действия, чем указание на рекомендацию, чтобы некое действие имело место.

Что касается шифрования медиапотоков в сочетании со вставкой полезной нагрузки, то этот текст в некоторых местах гласит: "Величина вставки должна определяться стандартным соглашением об алгоритме шифрования", см. например, 7.6.1, 8.3 и рисунок 1.7. Это означает, что некоторые алгоритмы шифрования (например, DES) обеспечивают дополнительные практические сведения о том как отправитель может выбрать величину бита(ов) заполнения. Примерами могут служить произвольные величины заполнения, статические величины или другие генерируемые кодовые комбинации. Какой бы метод ни применялся, это не скажется на способности к взаимодействию, даже при значительном отличии уровня защиты. Этот аспект касается проблем реализации и в дальнейшем в данной Рекомендации рассматриваться не будет.

6 Общие положения

6.1 Профиль защиты с шифрованием речевых сообщений

Профиль защиты с шифрованием речевых сообщений не является независимым профилем в отличие от базового профиля защиты. Это, скорее, опция вышеупомянутого профиля защиты, которая может использоваться совместно с ним. Этот профиль также основывается на определенных средствах защиты в качестве части процедур передачи сигналов вызова и установки соединения, например, согласования ключа Диффи-Хеллмана и других функций управления ключами.

Объекты H.323 могут использовать данную Рекомендацию для обеспечения конфиденциальности речевых сообщений. Предлагается четыре алгоритма шифрования: предложенные схемы представляют собой шифрование с использованием AES, RC2-совместимые, DES или тройные DES на основе бизнес-модели и требований к экспортируемости. В дополнение к режиму шифрования CBC, объекты H.323 могут реализовывать режим поточного шифрования EOFB. В некоторых случаях, когда уже обеспечена определенная степень конфиденциальности, могут не потребоваться шифрование речевых сообщений. В этом случае нет необходимости в согласовании ключей по алгоритму Диффи-Хеллмана или в других процедурах управления ключами.

Для дополнительной конфиденциальности речевых сообщений предлагается схема шифрования с использованием AES-128, RC2-совместимая, DES или тройная DES на основе бизнес-модели и требований к экспортируемости. В некоторых случаях, когда уже обеспечена определенная степень конфиденциальности, может не потребоваться шифрования речевых сообщений. В этом случае нет также необходимости в согласовании ключей по алгоритму Диффи-Хеллмана или в других процедурах управления ключами.

В данной Рекомендации рассматривается список возможных вариантов алгоритмов шифрования речевых сообщений, предлагавшихся в Рек. МСЭ-Т H.235, версия 2, Приложение D, или Рек. МСЭ-Т H.235, версия 3, Приложение D.

ПРИМЕЧАНИЕ 1. – Данный новый профиль алгоритма шифрования учитывает все известные сведения по криптоанализу и защите о стойкости алгоритмов шифрования и изменении в политике "криптоэкспорта". В частности, рассмотрение данной Рекомендацией алгоритмов шифрования учитывает требования к способности к взаимодействию с системами, соответствующими H.235 версии 2 или 3.

Объекты H.323, реализующие данную Рекомендацию с H.235 версии 4 или выше, должны предлагать 128-битовый AES в качестве предпочтительного алгоритма шифрования речевых сообщений в предоставляемых ими возможностях обеспечения защиты, чтобы добиться наивысшего качества и максимальной защиты. Такие объекты H.323 могут также дополнительно или по желанию предоставлять 168-битовый тройной DES в качестве алгоритма шифрования речевых сообщений, чтобы обеспечить бóльшую способность взаимодействия с системами H.323, в которых реализованы функции шифрования речевых сообщений из Приложения D/H.235 версии 2 или 3. Начиная с 56-битового DES и 56-битового совместимого RC2 алгоритмы шифрования не считаются в достаточной степени обеспечивающими защиту, объектам H.323 не следует предлагать эти слабые алгоритмы шифрования, если это не продиктовано острой необходимостью, такой как, например, обеспечение способности взаимодействия с системами шифрования речевых сообщений в Приложении D/H.235 версии 2 и 3.

Объекты H.323, реализующие данную Рекомендацию с H.235 версии 4 или выше, должны отдавать предпочтение предложенному 128-битовому AES, если это разрешено их политикой защиты. Такие объекты H.323 могут также дополнительно принимать 168-битовый тройной DES, если AES не предлагался или в противном случае не был разрешен их политикой защиты. Таким объектам H.323 не следует принимать 56-битового DES и 56-битового совместимого RC2 из соображений защиты, если только их политика защиты однозначно не одобряет такие незащищенные алгоритмы шифрования, или требования к экспорту требуют таких алгоритмов, а другие варианты с более высокой защитой, такие, как 128-битовый AES или 168-битовый тройной DES не предлагаются.

Средства управления доступом подробно не описываются; они могут быть реализованы локально после получения информации, переданной в сигнальных полях H.235 (ClearToken, CryptoToken).

В данной Рекомендации не описываются процедуры присвоения на основе подписки паролем/секретным ключом, а также контроля и административного управления. Такие процедуры могут проводиться средствами, выходящими за рамки действия данной Рекомендации.

Объекты, вовлеченные в процесс связи, могут неявно определять использование либо базового профиля защиты, либо профиля защиты в виде подписи путем оценки сигналов, передаваемых в сообщениях (**tokenOID** и **algorithmOID**, см. также пункт 11) идентификаторов объектов защиты.

В таблица 1 содержится краткое резюме функций защиты профиля шифрования речевых сообщений. Профиль шифрования речевых сообщений описывается в пунктах 7, 8 и 9.

Таблица 1/H.235.6 – Профиль шифрования речевых сообщений

<table>
<tr><th rowspan="2">Сетевые средства защиты</th><th colspan="4">Функции вызова</th></tr>
<tr><th>RAS</th><th>H.225.0</th><th>H.245</th><th>RTP</th></tr>
<tr><td>Аутентификация и целостность</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Неотказуемость</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Конфиденциальность</td><td></td><td></td><td></td><td>56-битовый DES
56-битовый RC2-совместим
168-битовый 3DES
128-битовый AES
Режим CBC или режим EOFB</td></tr>
<tr><td>Управление доступом</td><td></td><td></td><td></td><td></td></tr>
<tr><td>Управление ключами</td><td></td><td>Обмен аутентифицированными ключами Диффи-Хеллмана</td><td>Управление интегрированными сеансовыми ключами H.235 (Обмен аутентифицированными ключами Диффи-Хеллмана, обновление ключей)</td><td></td></tr>
</table>

Общая процедура формирует общий "ключ" (обмен Диффи-Хеллмана) между двумя участниками сеанса связи при инициировании соединения. Затем этот общий "ключ" используется для защиты (набора) медиаключей, которые используются для шифрования сеансов медиасвязи (RTP).

Профиль защиты на основе шифрования речевых сообщений является факультативным вариантом расширения для базового профиля защиты и для профиля защиты на основе подписи, его использование может быть оговорено при согласовании характеристик защиты терминала. В условиях, когда конфиденциальность речевых сообщений обеспечивается другими средствами, нет необходимости применения шифрования медийной информации и соответствующие процедуры управления ключами (согласие по ключам Диффи-Хеллмана, обновление ключей и синхронизация).

К выбранным алгоритмам шифрования относятся – AES, RC2-совместимый, DES и тройной DES.

ПРИМЕЧАНИЕ 2. – Так как вариант тройного DES может быть также использована для алгоритма DES, то это приводит к компактной реализации.

В независимости от выбора конкретного алгоритма шифрования медийной информации, необходимо четкое соответствие следующим опциям:

– Вектор инициализации (IV) генерируется, при необходимости, как указано в 9.3.1.

– Заполнение, при необходимости, должно производиться, как описано в 9.3.2.

Полезная аудионагрузка должна шифроваться с использованием согласованного алгоритма шифрования ("X1", "Y1", "Z1" или "Z2"), согласно процедурам, описанным в пункте 9 и 9.3, и методам заполнения шифротекста, описанным в пункте I.1. Полезная аудионагрузка может

шифроваться с использованием согласованного алгоритма шифрования ("X1", "Y1", "Z1" или "Z2"), действующего в режиме поточного шифрования (EOFB).

7 Сигнализация и процедуры H.245

В общем случае аспекты секретности медиаканалов контролируются так же, как любой другой параметр кодирования; каждый терминал указывает свои возможности, источник данных выбирает формат для использования, а получатель подтверждает или отклоняет этот режим. Все независимые от транспорта аспекты механизма, такие как выбор алгоритма, указываются в родовых элементах логического канала. Спецификации транспортных средств, таких как синхронизация с использованием алгоритма ключа/шифрования, передаются в специальных транспортных структурах.

7.1 Работа канала H.245

Предполагая, что процедуры соединения указывают на защищенный режим работы, то согласованное квитирование и аутентификация для канала управления H.245 должны проводиться до того, как произойдет обмен какими-либо отличными от H.245 сообщениями. Если согласовано, любой обмен сертификатами должен проводиться с использованием любого механизма, подходящего для терминала(ов) серии H. После завершения установления защиты канала H.245, терминалы используют протокол H.245 таким же образом, как и в режиме без защиты.

7.2 Работа канала H.245 без защиты

В ином случае канал H.245 может работать в режиме без защиты, тогда два объекта открывают защищенный логический канал, через который производится аутентификация и/или извлечение общего "ключа". Например, TSL (RFC 2246, RFC 3546) или IPsec (RFC 2401) могут использоваться посредством открытия логического канала с **dataType**, содержащем значение для **h235Control**. Этот канал затем может использоваться для получения общего "ключа", защищающего любой ключ медийного сеанса, или для транспортировки **EncryptionSync**.

7.3 Обмен характеристиками

Согласно описанным в 5.2/H.245 (Процедуры обмена характеристиками) и соответствующей Рекомендации по системам серии H, конечные точки обмениваются характеристиками с использованием сообщений H.245. Эти наборы характеристик могут в настоящее время содержать определения, которые указывают параметры защиты и шифрования. Например, конечная точка может обладать возможностью передавать и принимать видеосигналы H.261. Она также может сигнализировать о возможности передавать и принимать зашифрованные видеосигналы H.261.

Каждый алгоритм шифрования, используемый в совокупности с конкретным медийным кодеком, подразумевает определение новой характеристики. Как и в для любой другой характеристики, конечные точки при обмене могут сообщать как о независимых, так и о зависимых зашифрованных кодеках. Это позволяет конечным точкам изменить своих защитные характеристики в зависимости от дополнительных данных и имеющихся ресурсов.

После того как обмен характеристиками завершен, конечные точки могут открыть защищенные логические каналы для медийной информации таким же образом, как они бы это сделали при незащищенном режиме.

7.4 Роль ведущего объекта

Схема ведущий-ведомый H.245 используется для создания роли ведущего объекта с целью обеспечения работы канала в обоих направлениях и для решения других конфликтов. Эта роль ведущего используется также в методах защиты. Хотя режим(ы) защиты медиапотока устанавливается источником (в отличие от характеристик получателя), ведущий – это та конечная точка, которая формирует ключ шифрования. Такое создание ключа шифрования производится в независимости от того, является ли ведущий получателем или источником зашифрованной медийной информации. Чтобы обеспечить работу многовещательного канала с общими ключами, MC (также ведущий) должен формировать такие ключи.

7.5 Сигнализация логических каналов

Конечные точки открывают защищенные медийные логические каналы точно так же, как они открывают незащищенные медийные логические каналы. Каждый канал может действовать в полной независимости от других каналов, особенно, если это имеет отношение к защите. Конкретный режим должен быть определен в поле **OpenLogicalChannel dataType**. Исходный ключ шифрования должен передаваться либо в **OpenLogicalChannel**, либо в **OpenLogicalChannelAck**, в зависимости от соотношения ведущий/ведомый отправителя **OpenLogicalChannel**.

OpenLogicalChannelAck должен служить подтверждением режима шифрования. Если **openLogicalChannel** неприемлем для получателя, то или **dataTypeNotSupported**, или **dataTypeNotAvailable** (условие перехода) должны быть возвращены в поле причины **OpenLogicalChannelReject**.

Во время обмена протоколами, устанавливающего логический канал, ключ шифрования должен передаваться от ведущего к ведомому (независимо от того, кто является инициатором **OpenLogicalChannel**). Для медиаканалов, открытых другой конечной точкой (не являющейся ведущей), ведущий должен возвратить исходный ключ шифрования и исходную точку синхронизации в **OpenLogicalChannelAck** (в поле **encryptionSync**). Для медиаканалов, открытых ведущим, **OpenLogicalChannel** должен включать исходный ключ шифрования и точку синхронизации в поле **encryptionSync**.

7.6 Защита быстрого соединения

Конечные точки могут использовать процедуру быстрого соединения (см. 8.1.7 и 8.1.7.1/H.323), используя элемент быстрого старта для защищенного обмена данными ключей (задающего ключа и сеансовых ключей шифрования). Процедуры, указанные в 7.6.1, описывают "ровный" быстрый старт, который не использует множество предлагаемых алгоритмов шифрования, в то время как 7.6.1.1 описывает конкретный пример быстрого старта с использованием множества предлагаемых алгоритмов шифрования, делающими возможным более компактное кодирование сообщений.

7.6.1 Защита однонаправленного быстрого старта

Эта процедура описывает, как установить (полудуплексный) однонаправленный защищенный логический канал от вызывающей к вызываемой стороне.

Процедуры вызывающей стороны

Вызывающий объект (источник сообщения SETUP) объявляет как свой маркер DH, так и поддерживаемые им структуры FastStart. Маркер DH должен быть передан через встроенный ClearToken, как часть CryptoToken, или в качестве отдельного ClearToken, см. также 7.8. Во время передачи последовательности SETUP-to-CONNECT должен быть проведен обмен маркерами Диффи-Хеллмана (DH): это обеспечивает обе конечные точки общим "ключом". Поле **ClearToken** полей **CryptoToken** должно содержать **dhkey**, используемый для передачи параметров, как указано в этой Рекомендации. **halfkey** содержит произвольный открытый ключ одной из сторон, **modsize** содержит исходный ключ DH, а **generator** содержит DH-группу. Параметры DH, требующие применения, указана в таблице 4. За более подробной информацией следует обращаться к [RFC2412], Приложение E2.

ПРИМЕЧАНИЕ 1. – С момента аутентификации сообщений H.225.0 (как описано ранее в процедуре I), обмен DH аутентифицирован.

В любом направлении с сообщением сигнализации вызова, несущим полуключи Диффи-Хеллмана, если возможна идентификация информации, вызывающая или вызываемая сторона во время регистрации вызова должна также включать отдельный сквозной **ClearToken** с набором **sendersID**, установленным в идентификатор конечной точки отправителя и **tokenOID**, установленным в "E". Любой промежуточный объект сигнализации H.323 должен передавать этот сквозной маркер в неизменном виде.

Структуры FastStart блокируют предоставленные открытые логические каналы с предложенными характеристиками защиты. Следует предложить оба канала H235Cap и nonH235Cap. Во время обмена характеристиками H.245 Cap, конечные точки предоставляют входные данные **H235SecurityCapability** для кодеков, которые они поддерживают. Каждому кодеку соответствует

отдельная характеристика защиты H.235. Согласно таблице 6, эти характеристики должны указывать на поддержку 128-битового AES-CBC (OID – "Z3"), 56-битового RC2 – совместимого CBC (OID – "X"), они также должны указывать на поддержку 56-битового DES-CBC (OID – "Y") и могут указывать на поддержку 168-битового тройного DES-CBC (OID – "Z"), или 168-битового тройного DES-EOFB (OID – "Z1"), RC2-совместимого EOFB (OID – "X1"), DES-EOFB (OID – "Y1") или AES-EOFB (OID – "Z2").

(OpenLogicalChannel) передает как (**forwardLogicalChannelParameters**, так и **reverseLogicalChannel Parameters)** с **dataType**, обеспечивая элемент **h235Media** с (**encryptionAuthenticationAndIntegrity**, где в (**encryptionCapability** должен быть представлен, по крайней мере, один **MediaEncryptionAlgorithm**.

В целях взаимосвязи для защиты вызываемая сторона *априорно* является ведущей, см. также 7.4.

Вызывающая сторона должна установить истинное значение **mediaWaitForConnect**, чтобы удостовериться, что данные сеансового ключа доступны, и полученные зашифрованные медиаданные могут быть дешифрованы. В случаях, когда необходимо получение "досрочных", так называемых, медиаданных и вызываемая сторона одновременно передает зашифрованные или незашифрованные медиаданные с отправлением сообщения-ответа и данных ключей шифрования, вызывающая сторона должна быть готова к тому, что она не сможет дешифровать содержание, пока не будут доступны данные ключа.

ПРИМЕЧАНИЕ 2. – В случае, если вызываемая сторона посылает зашифрованные медиаданные вызывающей стороне (что теоретически она способна сделать, имея адреса RTP/RTCP вызывающей стороны), вызывающая сторона не сможет дешифровать их без общего "ключа", содержащегося в сообщении Connect (Alerting, Call Proceeding).

Процедуры вызываемой стороны

Во время FastStart вызываемая сторона представляет свой маркер DH (см. также 7.8) и приемлемые структуры FastStart. В случае применения процедуры Диффи-Хеллмана рекомендуется, чтобы вызываемая сторона возвращала свой маркер DH как часть сообщения ответа при первой же возможности, т. е. сообщения ответа, следующего непосредственно за SETUP. Это позволит вызывающей стороне рассчитать основной ключ из общего "ключа" DH и подготовиться к получению сеансового ключа и зашифрованных медийных данных.

ПРИМЕЧАНИЕ 3. – В случае если обеим сторонам недоступен алгоритм шифрования, медиапоток может остаться незашифрованным или может произойти разрыв соединения, в зависимости от стратегии защиты.

Каждый объект должен выделять соответствующие наименее значимые биты из общего "ключа" Диффи-Хеллмана для основного ключа шифрования (ведущий ключ); т. е. 56 наименее значимых битов "ключа" Диффи-Хеллмана для OID "X", OID "X1", OID "Y1" или OID "Y" и 168 наименее значимых битов "ключа" Диффи-Хеллмана для OID "Z", OID "Z1" или OID "Z2" и 128 наименее значимых битов "ключа" Диффи-Хеллмана для OID "Z3" или OID "Z2", см. также таблицу 6.

Ответы **OpenLogicalChannel(Ack)** выдаются с помощью сформулированного (основного) сеансового ключа, включенного в поле **encryptionSync**. Это **encryptionSync** сохраняет сеансовый ключ для направленного логического канала от вызывающей к вызываемой стороне. Транспортировка ключа должна осуществляться согласно процедуре, описанной в 8.3, с использованием либо **KeySyncMaterial,** либо **V3KeySyncMaterial** (см. 8.3.1). Сеансовый ключ должен быть зашифрован с помощью общего "ключа" DH, описанным ниже образом.

ПРИМЕЧАНИЕ 4. – Нет предписаний для формирования сеансовых ключей, которые используются для шифрования медиаданных. Формирование их значений зависит от реализации, на которую влияют местные ресурсы, стратегия и используемый алгоритм шифрования. Нужно соблюдать осторожность, чтобы избежать формирования нестойких ключей.

Используя процедуру 8.3, зашифрованный сеансовый ключ должен быть перенесен в **H.235Key**/**sharedSecret** в рамках поля **encryptionSync**. Сеансовый ключ должен быть перенесен в поле **keyMaterial** структуры **KeySyncMaterial**, если он не кратен размеру блока, то перед шифрованием он должен быть добавлен в массив блоков. Величина этого дополнения должна определяться посредством обычного согласованного алгоритма и шифрования. (Дополненная структура **KeySyncMaterial** должна шифроваться с использованием:

– 56 битов общего "ключа", начиная с наименее значащих битов из "ключа" Диффи-Хеллмана для OID "X", OID "X1" , OID "Y1" или OID "Y";

– всех битов общего "ключа" для OID "Z2", OID "Z" или OID "Z1", начиная с наименее значащих битов из "ключа" DH.

или же более предпочтительный вариант транспортировки улучшенного ключа, согласно 8.3.1, следует использовать там, где это возможно, благодаря процедуре индикации, версии 3 (см. 8.2).

В случае если, используя быстрый старт из двух однонаправленных каналов, должен быть создан полнодуплексный защищенный медиаканал, вызываемая сторона должна открыть второй логический канал в направлении вызывающей стороны. Этот логический канал должен передавать сигналы в отдельном элементе fastStart. Используя имеющийся общий "ключ" DH в качестве основного ключа, вызываемая сторона включает другой сеансовый ключ для этого логического канала в **encryptionSync**.

7.6.1.1 Использование множества алгоритмов шифрования при быстром соединении

Согласование алгоритмов шифрования медиаданных, как часть процедур быстрого соединения, ведет к необоснованному расширению числа элементов **OpenLogicalChannel** в элементе **fastConnect** сообщения SETUP. Это вызвано тем, что для каждой комбинации кодек (**dataType**) – алгоритм шифрования (в том числе, и "отсутствие алгоритма") требуется отдельный **OLC**.

Алгоритм шифрования, который должен быть применен к медиапотоку, устанавливается через включение структуры **dataType.h235Media.encryptionAuthenticationAndIntegrity.encryption Capability dataType** в **OLC**. В H.235v2 практикуется включение только одного **MediaEncryptionAlgorithm** в **encryptionCapability**, хотя этот последний элемент определяется как последовательность предшествующих элементов. Эта процедура делает возможным включение упорядоченной по приоритетам последовательности характеристик шифрования в каждом предлагаемом **OLC**. Получатель **OLC** должен затем выбрать один из предлагаемых алгоритмов и должен вернуть **OLC** с единственным выбранным алгоритмом (вместе с соответствующими транспортными адресами и информацией о ключах шифрования).

Для того, чтобы обеспечить максимальную эффективность ID объекта "NULL-ENCR" (см. таблицу 2) предоставляет алгоритм шифрования "нуль", что означает, что операция шифрования проводиться не будет. Использование этого конкретного метода требует наличия только одного **OLC** на каждый предлагаемый кодек и на каждое направление.

Таблица 2/H.235.6 – Идентификатор объекта для NULL-шифрования

Эталонное значение идентификатора объекта	Значение идентификатора объекта	Описание
"NULL-ENCR"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 26}	Указывает на "Алгоритм шифрования NULL"

Процедуры для вызывающей стороны (см. 8.1.7.1/H.323)

Если предлагаемый элемент **dataType** описывает шифрование через выбор **h235Media**, то включенный элемент **encryptionAuthenticationAndIntegrity** может включать элемент **encryptionCapability**, содержащий множество алгоритмов шифрования (включая алгоритм NULL). Эта конструкция должна применяться при предоставлении выбора любого из определенных алгоритмов для шифрования соответствующей медийной информации.

Процедуры для вызываемой стороны (см. 8.1.7.1/H.323)

Если для канала предлагается множество алгоритмов шифрования, то вызываемая конечная точка должна выбрать один из них и изменить **OpenLogicalChannel** для удаления остальных.

7.6.2 Защита двунаправленного быстрого старта

Защита для двухстороннего канала передачи данных T.120 находится на стадии разработки.

7.7 Зашифрованные DTMF сигналы H.245

Для достижения конфиденциальности конечные точки могут выбрать отправку шифрованных DTMF сигналов (RFC 2833). Используя сеансовый ключ шифрования, конечные точки могут зашифровать DTMF сигналы (RFC 2833) в **UserInputIndication** как:

- Зашифрованная основная строка: **encryptedAlphanumeric**;
- Зашифрованная строка iA5: **encryptedSignalType** в рамках **signal**;
- Зашифрованная общая строка: **encryptedAlphanumeric** в рамках **extendedAlphanumeric**.

ПРИМЕЧАНИЕ 1. – Дополнительные параметры для RTP в строке iA5 с отметками времени и номерами логических каналов или обновленные сигналы с длительностью тональных сигналов не шифруется, т. к. считается, что они не передают критическую информацию.

Согласованная характеристика **secureDTMF** относится к шифрованной строке iA5.

Управление ключами, как указано в пункте 6.1, следует применять, чтобы получить сеансовый ключ шифрования. Этот сеансовый ключ шифрования должен использоваться для шифрования DTMF сигналов H.245 (RFC 2833).

ПРИМЕЧАНИЕ 2. – Это необязательно означает, что сеансовый ключ должен применяться и для шифрования полезной нагрузки RTP.

Однако, когда DTMF (RFC 2833) также используется через RTP посредством установки флага **rtpPayloadIndication**, настоятельно рекомендуется защита полезной нагрузки RTP с использованием профиля шифрования речевых сообщений из 6.1.

В таблице 3 отображены имеющиеся алгоритмы шифрования (DES, 3DES or AES), которые должны использовать EOFB (включая OFB как особый случай, см. 8.4). Чтобы избежать потенциальной вставки символов DTMF (RFC 2833), не рекомендуется использовать для шифрования сигналов DTMF (RFC 2833) CBC, CFB или другие режимы формирования последовательности блоков, которые могут потребовать вставки.

7.7.1 Зашифрованная основная строка

Если в **UserInputCapability** выбран элемент **encryptedBasicString**, то тогда **encryptedAlphanumeric** должен указать примененный алгоритм шифрования в рамках **algorithmOID**, а **paramS** содержит исходное значение для операции шифрования. Зашифрованная буквенно-цифровая строка должна быть помещена в **encrypted**.

7.7.2 Зашифрованная строка iA5

Если в **UserInputCapability** выбран элемент **encryptedIA5String**, то тогда **encryptedSignalType** должен содержать шифрованный **ClearSignalType,** где **sig** переносит символ **signalType** нешифрованного текста. Элемент **signalType** должен содержать фиктивный "!", который должен быть аннулирован получателем.

Элемент **algorithmOID** должен указывать примененный алгоритм шифрования, **paramS** содержит исходное значение для операции шифрования.

7.7.3 Зашифрованная общая строка

Если в **UserInputCapability** выбран элемент **encryptedGeneralString**, то тогда элемент **encryptedAlphanumeric** в рамках **extendedAlphanumeric** должен указать примененный алгоритм шифрования в рамках **algorithmOID**, в то время как **alphanumeric** должен содержать пустую строку, а **paramS** содержит исходное значение для операции шифрования.

7.7.4 Список идентификаторов объектов

Таблица 3/H.235.6 – Идентификаторы объектов для шифрования DTMF сигналов H.245

Эталонное значение идентификатора объекта	Значение идентификатора объекта	Описание
"DES-EOFB-DTMF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 12}	Шифрование DTMF сигнала H.245 посредством DES-56 в режиме EOFB
"3DES-EOFB-DTMF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 13}	Шифрование DTMF сигнала H.245 посредством 3DES-168 в режиме EOFB
"AES-EOFB-DTMF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 14}	Шифрование DTMF сигнала H.245 посредством AES-128 в режиме EOFB

7.8 Функционирование системы Диффи-Хеллмана

Эта Рекомендация поддерживает протокол Диффи-Хеллмана для сквозного согласования ключей. В зависимости от ситуации согласованный ключ Диффи-Хеллмана может выступать в качестве основного ключа (см. 6.1) или в качестве динамического сеансового ключа (Рекомендации МСЭ-Т H.235.3 и H.530).

Система Диффи-Хеллмана характеризуется параметрами системы g и p, где p – универсальный основной ключ, а g обозначает генератор мультипликативных групп по модулю p или подгрупп с высоким уровнем по модулю p. $g^x \bmod p$ обозначает (открытый) полуключ Диффи-Хеллмана вызывающей стороны, в то время как $g^y \bmod p$ обозначает (открытый) полуключ Диффи-Хеллмана вызываемой стороны. RFC 2412 содержит дополнительную базовую информацию и дает советы по выбору параметров защиты Диффи-Хеллмана.

В Рек. МСЭ-Т H.235.0 представлен экземпляр системы Диффи–Хеллмана с параметрами (g, p, g^x), кодированными в рамках **ClearToken,** где **dhkey** содержит **halfkey** $g^x \bmod p$ (или соответственно $g^y \bmod p$) для некоторого секретного произвольного x *(*соответственно *y)*, основной p в **modsize** и **generator** g. Особым случаем является триплет (0, 0, 0) или пустой **dhkey,** который не содержит какого-либо экземпляра DH, но должен использоваться для сигнализации неиспользования профиля шифрования речевых сообщений.

Часто параметры системы DH p и g являются постоянными для ряда приложений с четкими значениями, хотя конечные системы могут также выбрать свой собственный набор параметров. Вызываемая сторона должна знать о том, что нестандартные параметры DH могут обеспечить меньшую защиту, чем те параметры, которые, на первый взгляд; выглядят как правомочные; например, вызывающая сторона может выбрать не основной ключ или же g генерирует меньшую подгруппу. До тех пор пока всесторонняя проверка параметров на практике неосуществима, вопрос о том, принимать или отвергать подобные предложения, относится к области стратегии защиты вызываемой стороны.

Для фиксированных параметров системы DH поверхностное описание параметров посредством идентификатора объекта может потребовать более компактных кодированных сообщений, чем те, что включают буквальные значения. Элемент **ClearToken**, который переносит экземпляр DH с фиксированными, стандартизованными параметрами DH, может указывать экземпляр DH через DH-OID в поле **tokenOID;** если только **tokenOID** не используется для других целей (описанных в пункте 7/H.235. для выделенного элемента **CryptoToken**). Отправитель может дополнительно включить буквальные значения DH, но в этом нет необходимости.

В случае, если должны быть указаны несколько экземпляров DH, причем каждый посредством DH-OID, параметры DH в выделенном **CryptoToken** (которому посвящена H.235.1) должны быть опущены путем дальнейшего отсутствия **dhkey**, а все экземпляры DH должны быть затем перенесены в рамках отдельных **ClearTokens,** где элемент **tokenOID** содержит DH-OID, а **dhkey** может и далее отсутствовать; все остальные поля в рамках этого элемента **ClearToken** не должны использоваться.

ПРИМЕЧАНИЕ 1. – Это не предусматривает возможности передачи экземпляра DH в выделенном **CryptoToken** или других имеющихся **ClearTokens** путем явного включения значений параметров DH.

В случае, если нужно указать на нестандартный экземпляр DH, должен использоваться DH-OID "DHdummy" и нестандартные параметры группы DH должны быть четко представлены в **ClearToken**.

Вызывающая сторона может представить один или несколько **ClearTokens**, каждый из которых переносит другой экземпляр Диффи-Хеллмана. Вызывающей стороне следует предоставить столько экземпляров DH, сколько позволяет ее стратегия защиты. Это дает вызываемой стороне возможность выбора соответствующего экземпляра DH для ответа, тем самым увеличивая вероятность нахождения удачного общего набора параметров.

Вызываемая сторона должна выбрать и принять один экземпляр DH (если это вообще произойдет), который она выбирает из неупорядоченного набора экземпляров DH, предоставленных вызывающей стороной в сообщении SETUP. В случае если вызываемая сторона может выбрать такой экземпляр DH, удовлетворяющий ее потребностям в защите, ей не следует изменять предложенный экземпляр DH или возвращать тот, который не был послан вызывающей стороной. Устойчивость алгоритмов шифрования, доступных обеим конечным точкам (EP) во время соединения, должна соответствовать устойчивости, обеспечиваемой выбранным экземпляром DH, который возвращается вызываемой стороной; см. таблицу 4. Вызываемая сторона должна указать выбранный экземпляр DH в сообщении ответа.

В случае если вызываемая сторона отклоняет все предложения по соображениям безопасности или из-за недостаточных возможностей по обработке вызова, ей следует опустить **dhkey** в сообщении ответа.

Вызываемая сторона должна включить свой маркер DH в ответ SETUP-to-CONNECT. Вызываемая сторона может включить свой маркер DH в сообщение, немедленно следующее за сообщением SETUP, или она может включить маркер DH на более поздней стадии, но не позднее сообщения CONNECT.

ПРИМЕЧАНИЕ 2. – Есть несколько аспектов, которые надо принимать во внимание при решении вопроса, когда именно вызываемая сторона должна включить маркер(ы) DH во время ответов SETUP-to-CONNECT: время ответа, обрабатываемая вызываемой стороной нагрузка, возможность преждевременной передачи медиаданных и другие аспекты. Считается, что эти вопросы зависят от реализации.

По некоторым причинам, однако, некоторые маршрутизирующие GK могут не передавать вызывающей стороне все ответы SETUP-to-CONNECT. Таким образом, одно или несколько сообщений ответа о посылке вызова H.225.0, включая возможный маркер DH, могут быть сброшены и поступят к вызывающей стороне. В этом случае вызывающая сторона не сможет определить основной ключ DH и сеансовый(е) медиаключ(и). Чтобы избежать такой ситуации, вызываемой стороне следует всегда включать один и тот же маркер DH в каждое ответное сообщение SETUP-to-CONNECT.

В случае если DH-OID указывает на другой экземпляр DH, а не на тот, который в действительности передается в рамках **modsize** и **generator**, буквальные значения, передаваемые в рамках **modsize** и **generator**, должны иметь приоритет под маркером DH-OID. Для ответа вызываемая сторона должна заменить конфликтующий DH-OID статическим DH-OID, например, "DH1024," который соответствует **modsize** и **generator** или "DHdummy", если нет соответствующего DH-OID.

7.8.1 Запрос на пересмотр параметров DH в середине вызова

Контроллер доступа H.323 может запросить пересмотр параметров DH в середине вызова, используя процедуры, определенные в этом пункте. Такая процедура пересмотра может понадобиться при осуществлении согласования ключей DH между конечной точкой, уже подсоединенной к контроллеру доступа, и конечной точкой, которую надо подключить (см. рисунок 1). Процедура пересмотра параметров Диффи–Хеллмана необходима для поддержки некоторых дополнительных услуг. Все процедуры, определенные в этом пункте, должны иметь место только в случае, если конечные точки H.323 находятся в состоянии "Передающая сторона приостановила работу", определенном в 8.4.6/H.323.

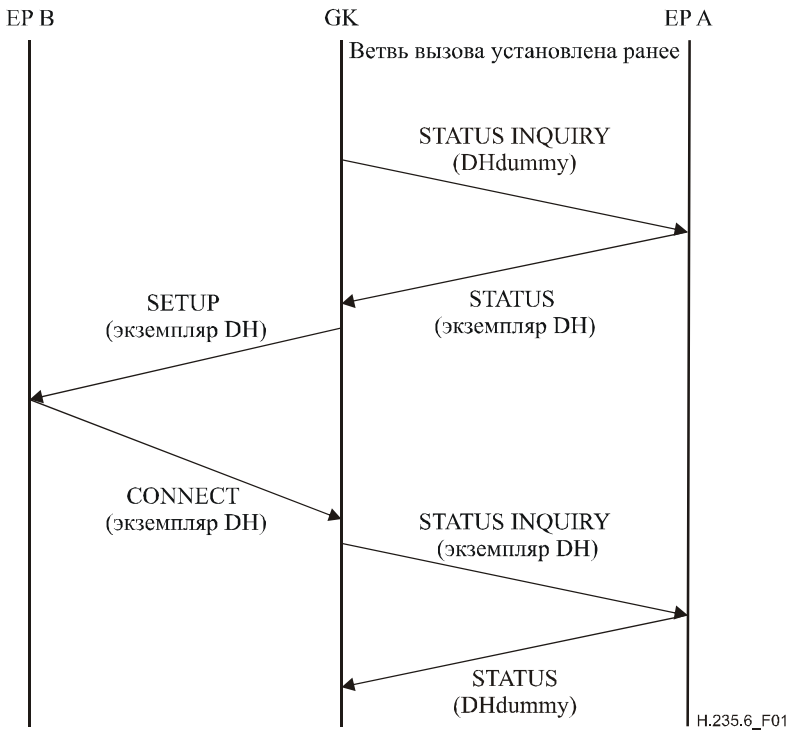


Рисунок 1/H.235.6 – Использование "Запрос параметров DH в середине вызова" для дополнительных услуг

Чтобы запросить параметры DH в середине вызова объект H.323 должен послать сообщение STATUS INQUIRY, содержащее поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID**, а остальные поля опущены.

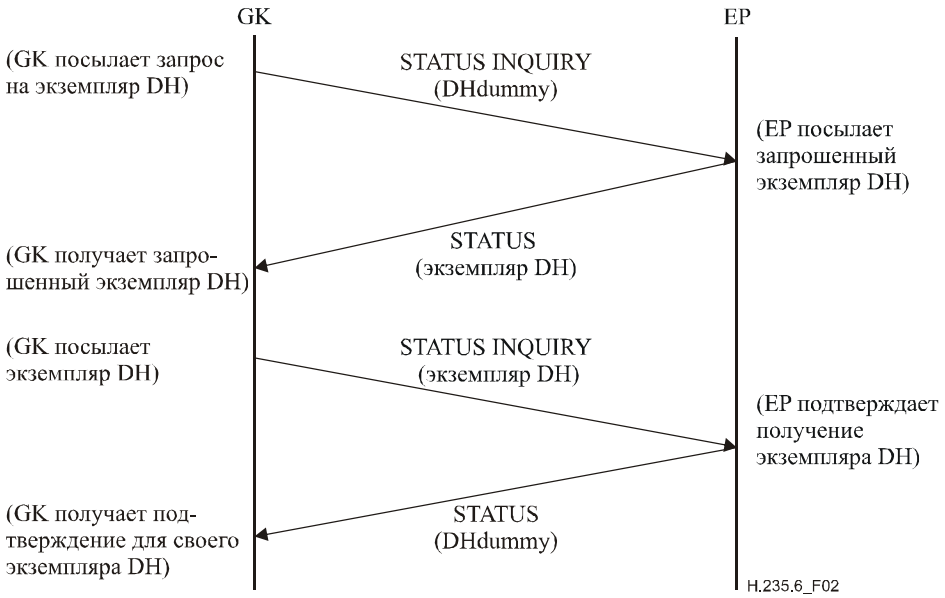


Рисунок 2/H.235.6 – Запрос параметров DH в середине вызова

Если объект H.323 получает сообщение STATUS INQUIRY, содержащее поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID,** конечная точка H.323 должна ответить сообщением STATUS, содержащим набор экземпляров DH, см. рисунок 2. Экземпляры DH должны быть указаны в этом сообщении STATUS, согласно правилам, определенным в 7.8 для сообщения SETUP.

ПРИМЕЧАНИЕ 1. – Предполагается, что объект H.323, который не поддерживает эту процедуру, отвечает на STATUS INQUIRY сообщением STATUS без экземпляров DH.

Для передачи принятого экземпляра DH в середине вызова объект H.323 должен послать STATUS INQUIRY, содержащий принятый экземпляр DH, см. рисунок 2. Экземпляры DH должны быть указаны в этом сообщении STATUS INQUIRY, согласно правилам, определенным выше, в 7.8. для ответов на сообщение SETUP.

Если конечная точка H.323 получает сообщение STATUS INQUIRY, содержащее поле **ClearToken** с экземпляром DH, конечная точка H.323 должна ответить сообщением STATUS, содержащим поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID**, а остальные поля опущены.

ПРИМЕЧАНИЕ 2. – Предполагается, что объект H.323, который не поддерживает эту процедуру, отвечает на STATUS INQUIRY сообщением STATUS без экземпляров DH.

Конечная точка H.323, получившая сообщение STATUS INQUIRY с экземпляром DH, должна пересчитать общий "ключ" DH на основе этого экземпляра DH и последнего набора экземпляра(ов) DH, присланного(ых) этой конечной точкой H.323 в конкретном вызове.

Если GK H.323 получает сообщение STATUS INQUIRY, содержащее поле **ClearToken** с экземпляром DH, или с DH-OID "DHdummy" в поле **tokenOID** тогда, за исключением нескольких случаев, приведенных ниже, он должен отправить это сообщение второй ветви вызова, считая от ветви, по которой было получено сообщение.

Если GK H.323 получает ответ STATUS на сообщение STATUS INQUIRY, которое он посылал, GK должен вернуть сообщение STATUS той ветви сообщения, по которой он получил сообщение STATUS INQUIRY.

Если GK H.323, ожидая ответ на сообщение STATUS INQUIRY, содержащее поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID,** которое он посылал, получает сообщение STATUS INQUIRY, содержащее поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID** и с флагом CVR, установленным на 1, то GK должен ответить сообщением STATUS, содержащим поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID** (см. рисунок 3).

Если GK H.323 получает сообщение STATUS INQUIRY, содержащее поле **ClearToken** с DH-OID "DHdummy" в поле **tokenOID,** в то время как вторая ветвь вызова отсутствует, GK должен ждать установки второй ветви вызова, послать по этой ветви соединения пустой набор возможностей и затем передать по этой ветви полученное сообщение STATUS INQUIRY (см. рисунок 3).

GK H.323 не должен инициировать никаких процедур, определенных в этом пункте после того, как он отправил сообщение STATUS, содержащее экземпляр DH и до того, как он получил сообщение STATUS INQUIRY, содержащее экземпляр DH.

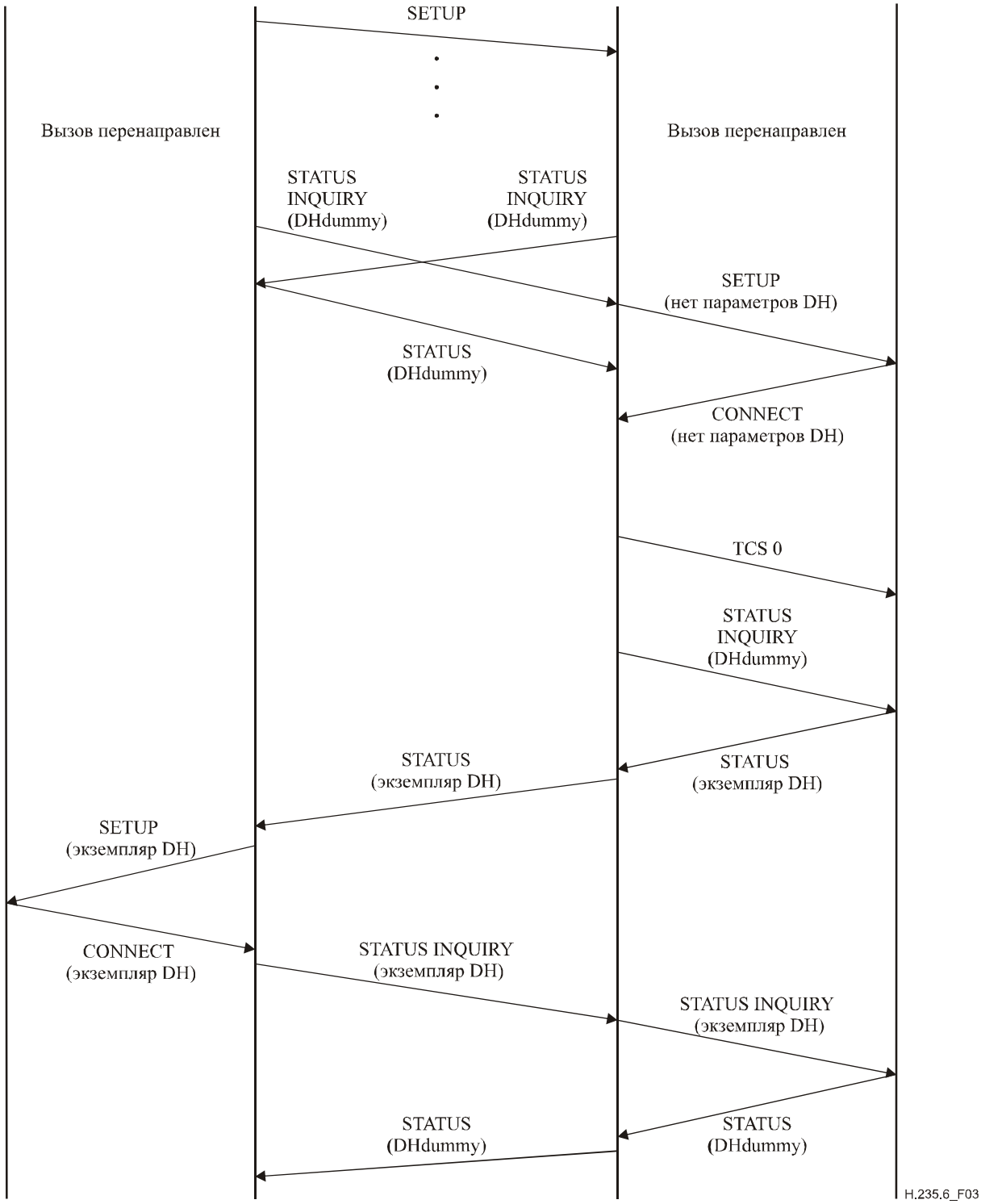


Рисунок 3/H.235.6 – Использование "Запрос параметров DH в середине вызова" для одновременного перенаправления вызова обоими GK

8 Передача сигналов и процедуры

Необходимо следовать процедурам, изложенным в пункте 8/H.323 (Процедуры передачи сигналов вызова). Конечные точки H.323 должны иметь возможность кодировать и распознавать наличие (или отсутствие) требований защиты (для канала H.245), передаваемых в сообщениях H.225.0.

В случае, когда надо обеспечить защиту самого канала H.225.0, необходимо следовать тем же самым процедурам из пункта 8/H.323. Различие в работе заключается в том, что процесс связи должен происходить только после соединения с защитным идентификатором TSAP и использования заранее

определенных режимов защиты (например, TLS (RFC 2246, RFC 3546)). Благодаря тому, что при установлении связи согласно H.323 сначала идет обмен сообщениями H.225.0, не может быть никакого внутриполосного согласования защиты в полосе для H.225.0. Другими словами, обе стороны должны заранее знать, что они используют конкретный режим защиты. Для H.323 в IP, для осуществления процесса связи для защиты TLS (RFC 2246, RFC 3546) используется альтернативный общеизвестный порт (1300).

Одна из целей обмена сообщениями H.225.0 в соответствии с защитой H.323 является обеспечение механизма для организации защищенного канала H.245. При желании аутентификация может проводиться при обмене сообщениями H.225.0. Эта аутентификация может основываться на сертификате или пароле, использовать шифрование и/или хеширование (т. е. подписывание). Особенности этих режимов работы описаны в пунктах с 8.1 по 8.2.3/H.235.0.

Конечная точка H.323, которая принимает сообщение SETUP с набором **h245SecurityCapability,** должна ответить соответствующим приемлемым **h245SecurityMode** в сообщении CONNECT. В случае если наложенных характеристик нет, вызываемый терминал может отказать в соединении, отправив **Release Complete** с кодом причины, установленным в ***SecurityDenied***. Эта ошибка не предназначена для передачи информации о несоответствии защиты, и вызывающий терминал должен искать другие средства для определения проблемы. В случаях, когда вызывающий терминал получает сообщение CONNECT без достаточного, или приемлемого, режима защиты, он может завершить вызов посредством **Release Complete** с ***SecurityDenied***. В случаях, когда вызывающий терминал получает сообщение CONNECT без каких-либо защитных характеристик, он может завершить вызов посредством **Release Complete** с ***undefinedReason***.

Если вызывающий терминал получает сообщение о приемлемом режиме **h245Security**, то он должен открыть и использовать канал H.245 в указанном режиме защиты. Безуспешное установление канала H.245 в режим защиты, определенным здесь, следует рассматривать как ошибку протокола и соединение блокируется.

8.1 Совместимость с Редакцией 1

Имеющая возможности защиты конечная точка не должна возвращать никакие относящиеся к защите поля, указания или статус обладающей возможностью защиты конечной точке. Если вызывающая сторона получает сообщение SETUP, которое не содержит характеристик **H245Security** и/или маркер аутентификации, то она может возвратить **ReleaseComplete,** для отклонения соединения; но в этом случае она должна использовать код причины ***UndefinedReason***. Подобным образом, если, отправив сообщение SETUP с **H245Security** и/или маркером аутентификации, вызывающая сторона получает сообщение CONNECT без **H245SecurityMode** и/или маркера аутентификации, то она может блокировать соединение, выдав **ReleaseComplete** с кодом причины ***UndefinedReason***.

8.2 Указание возможностей версии 3

Конечные точки H.235 версии 3 и более поздних версий обеспечивают улучшенные процедуры защиты медийного тракта, которые не поддерживаются H.235, версии 1 и 2. Такими улучшенными процедурами защиты являются:

- улучшенная транспортировка ключа (**V3KeySyncMaterial**, см. 8.3.1);
- улучшенное обновление ключа, см. 8.6.2.

Поскольку конечные точки не имеют представления о взаимной поддержке H.235, версии 3 или более поздних версий, то во время установки вызова добавляется точная индикация версии.

Конечным точкам H.235, версии 3 и более поздним версиям для определения возможностей версии 3 (улучшенная транспортировка ключа, улучшенная синхронизация шифрования) следует всегда использовать процедуры, описанные в этом пункте. В зависимости от результатов процедуры передачи логических сигналов, конечные точки могут использовать эти процедуры (см. 8.3) для обратной совместимости с конечными точками H.235 версии 1 или версии 2.

Чтобы указать, используются ли улучшенные процедуры H.235 версия 3, вызывающая и вызываемая конечные точки должны включать дополнительный **ClearToken,** показывающий во время сигнализации вызова (SETUP, CONNECT и т. п.) на возможность версии 3. Отсутствие такого **ClearToken** будет указывать на то, что поддерживается возможности только H.235 версии 1 или

версии 2. В этом случае конечная точка должна использовать процедуру из 8.3. В противном случае, конечная точка может использовать улучшенные процедуры, описанные в 8.3.1, или использовать процедуру 8.3 H.235 версии 1 или версии 2.

Этот **ClearToken** должен использовать **tokenOID**, установленный на "V3", и ему присваивается следующее значение.

"V3"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 24}	Индикатор возможностей версии 3 в ClearToken во время сигнализации вызова.

Все другие поля в этом **ClearToken** должны оставаться неиспользованными, если они не используются для передачи параметров DH.

8.3 Транспортировка ключей

Ведущий объект должен формировать данные сеансовых ключей и распространять их к одноранговому(ым) объекту(ам). Для транспортировки ключа предлагаются две процедуры:

- Процедура, рассчитанная в основном на конечные точки H.235 версии 1 или версии 2; описанная в данном пункте.
- Улучшенная процедура, рассчитанная на конечные точки H.235 версии 3 и более поздние версии, описанная в 8.3.1.

Конечные точки H.235 версии 1 или версии 2 применяют для транспортировки сеансовых ключей следующую процедуру:

KeySyncMaterial содержит идентификатор конечной точки ведущего объекта в рамках **generalID** и переносит данные сеансового ключа в рамках **keyMaterial**. Значение **generalID** должно быть включено для обеспечения минимального уровня аутентификации источника сеансового ключа (см. также 8.6). Получатель должен проверить правильность полученного **generalID**.

ПРИМЕЧАНИЕ. – Эта Рекомендация предполагает, что каждая конечная точка зарегистрирована контроллером доступа и обладает идентификатором конечной точки, который может быть передан в рамках **generalID**. Эта Рекомендация не содержит сценарий без контроллеров доступа; это остается областью для дальнейших изучений.

KeySyncMaterial должен быть зашифрован с использованием согласованного основного ключа. **KeySyncMaterial** должен всегда добавляться к множеству блоков перед шифрованием, при этом последнему октету должно быть задано значение, соответствующее количеству октетов заполнителей (включая последний). Значение заполнителя следует определять на основании обычного соглашения алгоритма шифрования. Результат шифрования должен будет сохраняться в **sharedSecret** ключа **H235Key**.

8.3.1 Улучшенная транспортировка ключа согласно H.235, версии 3

Замечено, что синтаксическое определение в синтаксисе ASN.1 элемента **KeySyncMaterial** и метод, согласно которому операция ENCRYPTED{} применяется к данным в H.235 версии 1 и 2, раскрывает большое количество известных нешифрованных текстов: прежде всего, **generalID** ведущего объекта, но, кроме того, некоторые известные биты кодирования для этой структуры. **generalID**, даже будучи зашифрован, известен из других нешифрованных частей сигнального сообщения (например, **senderID**). Считается, что наличие таких известных нешифрованных текстов значительно ослабляет схему защиты, поскольку атакующий объект может с большей легкостью "взломать" сеансовый ключ применением "грубой силы", особенно для блочного шифра, который имеет меньшую длину блоков, такого как DES-56 или RC2-совместимый.

Кроме того, H.235 версии 3 должна иметь возможность для транспортировки дополнительных данных ключа:

- обеспечить защиту транспортировки расширенного ключа к одноранговому(ым) объекту(ам). Такой расширенный ключ вводится для улучшенного режима OFB; см. 8.4.

H.235 версии 3 дополняет **H235Key** элементом **secureSharedSecret,** содержащим **V3KeySyncMaterial,** который включает следующие параметры:

generalID содержит идентификатор конечной точки инициирующего отправителя, если он имеется, иначе поле остается неиспользованным.

algorithmOID указывает на примененный алгоритм шифрования и режим работы.

paramsS содержит значение инициализации, которое применяется для шифрования переданного(ых) ключа(ей).

ПРИМЕЧАНИЕ 1. – Значение IV в рамках **paramS** нельзя смешивать с величиной IV каждого пакета RTP, который не передается. **ClearSalt** дополнительно содержит нешифрованный расширенный ключ для шифрования сеансовых ключей (например, для EOFB).

encryptedSessionKey содержит шифрованный текст зашифрованного исходного сеансового ключа.

encryptedSaltingKey содержит шифрованный текст зашифрованного исходного медийного расширенного ключа, если он существует. Расширенный ключ необходим для улучшенного режима OFB.

clearSaltingKey может содержать незашифрованный исходный медийный расширенный ключ. При применении необходимо убедиться, что **encryptedSaltingKey** и **clearSaltingKey** не используются одновременно.

paramSsalt содержит исходное значение для шифрования расширенного ключа. **ClearSalt** дополнительно содержит незашифрованный расширенный ключ для шифрования этого ключа (например, для EOFB).

ПРИМЕЧАНИЕ 2. – **generalID**, **algorithmOID** и **paramS** всегда передаются в незашифрованном тексте, в то время как **encryptedSessionKey**, **encryptedSaltingKey** содержат шифрованный текст с данными зашифрованного ключа.

Ведущий объект формирует ключ(и) согласно согласованным возможностям терминала и посылает ключ(и) к одноранговой(ым) конечной(ым) точке(ам), используя **V3KeySyncMaterial**. Таким образом, **V3KeySyncMaterial** при его наличии должен посылаться неизмененным промежуточными контроллерами доступа.

Конечные точки H.235 версии 3 или более поздних версий должны всегда использовать **secureSharedSecret** в рамках **H235Key,** но, в зависимости от выходных результатов процедуры сигнализации логического канала из 8.2, задействующей инициирующий **ClearToken** версии 3, они могут использовать **sharedSecret** для обратной совместимости с конечными точками H.235 версии 1 или версии 2.

8.4 Усовершенствованный режим OFB

Режим OFB (ИСО/МЭК 10116) определяет режим работы, который осуществляет потоковое шифрование, используя алгоритмы блочного шифрования. Режим OFB обеспечивает:

- улучшенное качество работы за счет сокращения задержки на проведение шифрования;
- более легкую и менее сложную обработку неполных блоков;
- хорошую устойчивость в отношении ошибок по битам.

Усовершенствованный режим OFB – это слегка модифицированный режим OFB, названный здесь "усовершенствованным режимом обратной связи по выходу" (EOFB), который имеет те же характеристики, что и OFB, но в дополнение к ним:

1) использует расширенный ключ KS в дополнение к ключу шифрования KE; и
2) вводит индекс неявного пакета.

Использование дополнительного секретного расширенного ключа KS, подвергнутого операции XOR ("исключающее" или) в обратном направлении, дает дополнительную защиту при обработке и анализе известных нешифрованных текстов. Это главное преимущество в части безопасности, которое другие стандартные режимы работы (такие как CBC, OFB и т. п.) не обеспечивают. Использование режима EOFB выражается, таким образом, в повышении стойкости защиты при обработке и анализе высоко избыточных нешифрованных текстов и общеизвестных нешифрованных текстов.

EOFB определяется как $C_i = P_i \oplus S_i$, где $S_i = E_{KE}(KS \oplus S_{i-1})$ для $i = 1 \ldots n$ и $S_0 = IV$, где C_i является i-тым блоком шифротекста, P_i является i-тым блоком нешифрованного текста, S_i является i-тым ключом потока ключей, KE – ключ шифрования и $\oplus$ побитовая XOR. EOFB проиллюстрирован на рисунке I.6.

EOFB может быть также выполняется в стандартном режиме OFB, совмещая обратные сообщения в EOFB с сообщениями в OFB. В случаях, когда необходима обратная совместимость со стандартным режимом OFB, расширенный ключ KS должен быть установлен на все нули приближенно к этому, либо оставляя незаполненным поле **encryptedSaltingKey** в рамках **V3KeySyncMaterial** пустым. Однако использование действующего расширенного ключа настоятельно рекомендуется в случаях, когда полезные сигналы RTP шифруются блочным шифром, имеющим более короткую длину блоков, таким как DES-56 или RC2-совместимый.

После обработки, по крайней мере, 2^{48} пакетов должен использоваться новый сеансовый ключ шифрования KE и новый расширенный ключ KS, в противном случае возникнет ситуация повторного использования потока ключей, что ставит под угрозу безопасность данных.

Пункт 11 определяет идентификаторы объектов для DES-56-EOFB, RC2-совместимого-EOFB, 3-DES-EOFB и AES-EOFB.

8.5 Управление ключами

Конечным точкам, отвечающим положениям данной Рекомендации, следует использовать процедуру быстрого соединения согласно 7.6.1. Если быстрый старт не применяется, то по данной Рекомендации для защиты сообщений управления вызовом должно использоваться туннелирование H.245. Процедуры быстрого старта делают возможным установку одного либо двух однонаправленных логических каналов. Процедура быстрого старта способствует согласованию характеристик защиты для распределения общего секретного секрета (общий ключ DH), действующего в качестве главного ключа, и для безопасного распределения ключа шифрования.

Таблица 4 содержит распределенные OIDs для различных алгоритмов шифрования и соотносит их с распределенными OIDs для группы Диффи-Хеллмана. Через OID определяются три группы DH:

- "DHdummy": экземпляр данной группы DH следует использовать, как только подразумевается обеспечение экспортируемой защиты (512 битов) или если используется любые из нестандартных групп DH.

  ПРИМЕЧАНИЕ 1. – Никакой конкретной группы DH не определяется; OID указывает на любую нестандартную группу DH.
- Экземпляр 512-битовой группы DH должен использоваться для создания главного ключа для распределения сеансового(ых) ключа(ей) для RC2-совместимого ("X") или для DES-56 битового алгоритма шифрования ("Y").
- "DH1024": Данная группа DH применяется, в случае если подразумевается обеспечение высокой (1024 битов) безопасности. OID указывает на стандартизованную, фиксированную группу DH. Данная группа DH должна быть использована для формирования главного ключа для распределения сеансового(ых) ключа(ей) для алгоритма шифрования тройной DES ("Z").
- "DH1536": Данная группа DH предлагается как вариант для конечных точек версии 3, имеющих очень высокие требования к защите, превышающие защиту 1024-битовой группы DH. OID указывает на фиксированную группу DH. Данная группа должна быть использована для формирования главного ключа для распределения сеансового(ых) ключа(ей) для алгоритмов шифрования тройной DES ("Z", "Z1") или AES-128 ("Z2", "Z3").

Рекомендуется применять указанные 1024-битовых или, что необязательно, 1536-битовые группы DH, если только другие требования защиты не приведут к предпочтению других параметров Диффи–Хеллмана. Далее, рекомендуется рассмотреть вопрос использования указанных OIDs, при идентифицировании групп DH, см. 7.8. Тем не менее при реализации следует быть готовым к получению параметров групп DH, без явной индикации OID. В таком случае при реализации следует удостовериться, что, согласно таблице 4, передается правильная группа DH.

Конечные точки могут использовать параметры нестандартных групп DH. Использование OID "DHdummy" должно указывать на такие нестандартные группы DH. Решение, принимать или нет такие группы DH, принимает вызываемая сторона.

ПРИМЕЧАНИЕ 2. – Выбор группы DH не отменяет необходимость в согласовании действующего алгоритма шифрования медиаданных. Это должно быть выполнено с помощью процедуры согласования характеристик терминала H.245.

ПРИМЕЧАНИЕ 3. – Во время установки соединения (SETUP-to-CONNECT) применение алгоритма шифрования OIDs не должно использоваться для указания экземпляра группы Диффи-Хеллмана.

Таблица 4/H.235.6 – Группы Диффи – Хеллмана

OID алгоритм шифрования	**DH-OID**	**Описание групп DH**
"X", "X1" (RC2-совместим.), "Y", "Y1" (DES)	"DHdummy"	Mod-P, любой подходящий 512-битовый исходный ключ
" Z ", " Z1 " (тройной DES), " Z2 ", " Z3 " (AES)	"DH1024"	Mod-P, 1024-битовый исходный ключ Исходный ключ = $2^{1024} – 2^{960} – 1 + 2^{64} × \{ [2^{894}$ pi] + 129093 } =(179769313486231590770839156793787453197860296048756011706444 423684197180216158519368947833795864925541502180565485980503 646440548199239100050792877003355816639229553136239076508735 759914822574862575007425302077447712589550957937778424442426 617334727629299387668709205606050270810842907692932019128194 467627007)$_{10}$ Генератор (Примечание) = 2
" Z ", " Z1 " (тройной DES), " Z2 ", " Z3 " (AES)	"DH1536"	Mod-P, 1536-битовый исходный ключ Исходный ключ = $2^{1536} – 2^{1472} – 1 + 2^{64} × \{ [2^{1406}$ pi] + 741804 } =(241031242692103258855207602219756607485695054850245994265411 694195810883168261222889009385826134161467322714147790401219 650364895705058263194273070680500922306273474534107340669624 601458936165977404102716924945320037872943417032584377865919 814376319377685986952408894019557734611984354530154704374720 774996976375008430892633929555996888245787241299381012913029 459299994792636526405928464720973038494721168143446471443848 852094012745984428885933652689632091963391 9)$_{10}$ Генератор (Примечание) = 2
ПРИМЕЧАНИЕ. – Генератор используется для создания DH маркера.		

8.6 Обновление и синхронизация ключей

Для 64-битовых блочных шифров частота обновления ключей *должна быть* такой, что не более 2^{32} блоков могут быть зашифрованы с использованием одного и того же ключа. Реализациям *следует* обновлять ключи до того, как одним и тем же ключом будут зашифрованы 2^{30} блоков (см. 9.1). Для 128-битовых блочных шифров частота обновления ключа *должна быть* такой, что не более 2^{64} блоков могут быть зашифрованы с использованием одного и того же ключа. Реализациям *следует* обновлять ключи до того, как одним и тем же ключом будут зашифрованы 2^{62} блоков (см. 9.1). Оба участвующих объекта могут менять медийный сеансовый ключ так часто, как это требует их стратегия защиты. Например, ведущий объект может распределять новый сеансовый ключ, используя **encryption Update** или **encryptionUpdateCommand** сообщения **miscellaneousCommand**. С другой стороны, ведомый объект может запросить у ведущего новый сеансовый ключ, используя **encryptionUpdateRequest** сообщения **miscellaneousCommand**.

Сообщение **MiscellaneousCommand** содержит **encryptionUpdate** и **encryptionUpdate Command,** исходя из которых устанавливаются следующие параметры **encryptionSynch**:

- **synchFlag**: новый динамический номер полезной нагрузки RTP, показывающий смену ключа.
- **h235key**: передает новый зашифрованный сеансовый ключ. Это закодированная в ASN.1 структура в **H235Key** H.235**,** передаваемая в виде цепочки объектов.

Поле **sharedSecret** в структуре **H235Key** использует следующие поля:

- **algorithmOID**: устанавливается на "X", "X1" для 56-битового RC2-совместимого алгоритма, устанавливается на "Y", "Y1" для 56-битового DES или устанавливается на "Z", "Z1" для 168-битового тройной DES или устанавливается на "Z3" для 128-битового AES.

  ПРИМЕЧАНИЕ 1. – Алгоритм шифрования сеансового ключа тот же, что и согласованный алгоритм шифрования медиаданных.
- **paramS**: устанавливается на исходное значение. Для 64-битовых блочных шифров **iv8** содержит случайную 64-битовую блочную комбинацию, формируемую инициатором. Для 128-битовых блочных потоковых шифров **iv16** содержит случайную 128-битовую блочную комбинацию, формируемую инициатором. Данное поле не должно использоваться в режиме CBC и должно быть установлено на NULL, означающий, что CBC-IV для шифрования сеансового ключа должен быть установлен на 0; он должен использоваться исключительно для переноса IV в режиме EOFB.
- **encryptedData**: устанавливается на результат зашифрованного **KeySyncMaterial**.

В качестве составной части **KeySyncMaterial**:

- **generalID**: идентификатор источника, распределяющего ключ.

  ПРИМЕЧАНИЕ. – Эта Рекомендация предполагает, что каждая конечная точка зарегистрирована контроллером доступа и получила идентификатор конечной точки, который может быть передан в рамках **generalID**. Эта Рекомендация не поддерживает сценарии без контроллеров доступа; это остается областью дальнейших изучений.
- **keyMaterial**: устанавливается на новый сеансовый ключ. Для DES и RC2-совместимого алгоритмов это – 56-битовый ключ, для тройного DES это – 168-битовый ключ, а для AES это – 128-битовый ключ. Ведущий должен сформировать новый сеансовый ключ, который отвечает по крайней мере следующим критериям защиты: это не должен быть неустойчивый или отчасти неустойчивый DES-ключ, и он должен использовать достаточно защищенный случайный источник.

Сообщение **MiscellaneousCommand** содержит **encryptionUpdateRequest,** который содержит **keyProtectionMethod,** где флаг **sharedSecret** установлен в TRUE.

ПРИМЕЧАНИЕ 3. – Поскольку обновление и синхронизация ключей зависят от сообщений H.245, которые обратно не совместимы во время быстрого соединения, то для защищенных объектов H.323 требуется туннелирование сообщений H.245.

Медийные сеансовые ключи не существуют вечно. В какой-то момент времени срок действия каждого сеансового ключа истекает. Тогда для защиты текущего сеанса безопасности следует использовать новый сеансовый ключ. В условиях конференции следует определить и распределить новый ключ группового сеанса в тот момент, когда участники группы присоединяются или покидают защищенную конференцию, тем самым, предупреждая получение ими предыдущих или последующих данных.

- Обновление и синхронизация ключей, основанных на типе полезной нагрузки, определяют новый тип динамического полезной нагрузки для этого нового сеансового ключа, см. пункты 8.6.1, 8.6.2 и 8.6.3.

Для обновления ключей в данной Рекомендация предлагается неподтвержденное квитирование, применимое также для конечных точек H.235 версии 1 и версии 2, а также устойчивое к ошибкам подтвержденное квитирование для конечных точек H.235 версии 3 и последующих версий.

8.6.1 Обновление неквитированных ключей

Рисунок 4 демонстрирует неподтвержденное квитирование для распределения/обновления сеансовых ключей. Если ведомому объекту необходим обновленный сеансовый ключ, он может запросить

новый сеансовый ключ у ведущего объекта, отправив ему сообщение **encryptionUpdateRequest**. Этот ведущий объект должен послать ведомому объекту новый сеансовый ключ (при наличии или отсутствии предварительного сообщений **encryptionUpdateRequest** от ведомого объекта) в рамках сообщения **EncryptionUpdate**.

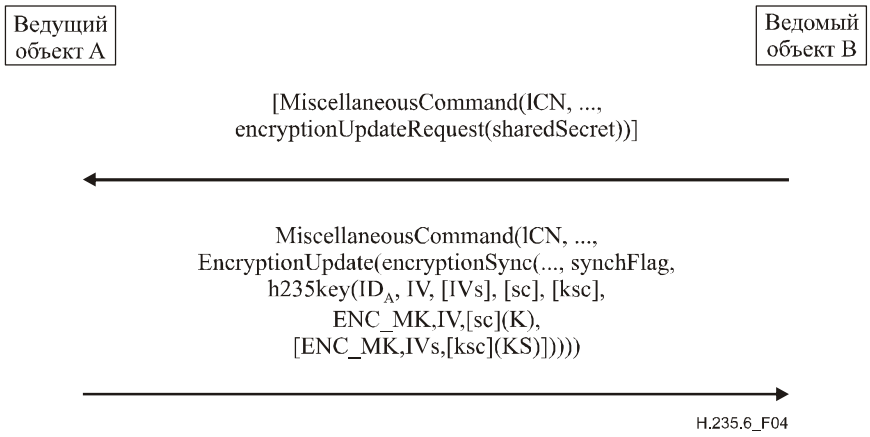


Рисунок 4/H.235.6 – Обновление и распределение неквитированного сеансового ключа от ведущего объекта к ведомому(ым)

где:

lCN	номер логического канала;
synchFlag	новый номер динамической полезной нагрузки RTP;
ID_A	**generalID** источника;
IV	исходное значение/вектор для шифрования сеансового ключа;
IVs	исходное значение/вектор для шифрования расширенного ключа;
ENC_M,IV,sc(K)	означает шифрование нешифрованного текста *K,* используя ключ *M*, исходный вектор *IV* [и расширенный ключ *sc*, только для EOFB];
KS	расширенный ключ для медиа (только для режима EOFB);
K	сеансовый ключ для нешифрованного текста;
sc	незашифрованный расширенный ключ, когда для шифрования сеансового ключа используется режим EOFB;
ksc	нешифрованный расширенный ключ, когда для шифрования расширенного ключа используется режим EOFB;
s2M/m2S	флаг **direction** (только для H.235v3 и последующих версий) (s2m = ведомый-ведущий, m2s = ведущий-ведомый);
[]	представляет необязательную часть.

Методы обновления ключей, как описывается в следующих пунктах, могут использовать режим шифрования EOFB для защиты данных передаваемых ключей. Чтобы использовать режим EOFB для защиты данных ключей таким же образом, как и при защите полезной нагрузки медиаданных, необходимо использовать дополнительный расширенный ключ (sc или ksc).

8.6.2 Улучшенное обновление ключей

Конечные точки H.235 версии 3 и последующих версий должны выполнять процедуру обновления явных/неявных квитированных ключей. Это сделано для обеспечения надежных методов обновления ключа, базирующихся на методе обновления неквитированных ключей, представленного в версиях до H.235v3. Возможности такой процедуры должны быть согласованы с использованием индикации возможностей версии 3, согласно 8.2.

Рисунок 5 отображает процедуры обновления ключей для логического канала, владельцем которого является ведомый объект. В случае если ведомый инициирует обновление ключа и запрашивает у ведущего объекта новый сеансовый ключ, ведомый объект должен послать ведущему объекту **MiscellaneousCommand**, где элемент **logicalChannelNumber** должен содержать номер логического канала (его определяет ведомый), элемент **sharedSecret** должен быть установлен на истинное значение, флаг **direction** должен быть установлен на **slaveToMaster** и номер новой динамической полезной нагрузки должен быть запрошен в **synchFlag** в рамках **EncryptionUpdateRequest**. Если,

напротив, ведущий объект инициирует обновление ключей, сообщение **EncryptionUpdateRequest** не должно посылаться.

Ведущий объект, или отвечая на запрос ведомого объекта, или от своего имени, должен выдать команду **Encryption UpdateCommand,** где элемент **logicalChannelNumber** должен содержать номер логического канала, флаг **direction** должен быть установлен на **slaveToMaster** в рамках **MiscellaneousCommand,** а элемент **synchFlag** в рамках **encryptionSync** отражает номер новой динамической полезной нагрузки.

Элемент **h235key** должен переносить новый сеансовый ключ. **h235key** должен содержать идентификатор ведущего объекта в **generalID** и примененный исходный вектор *IV* в **paramS**. Шифрованный медийный сеансовый ключ должен быть передан в рамках **encryptedSessionKey**, где функция шифрования должна применять сеансовый ключ ведущего объекта и исходное значение в **paramS** к сеансовому ключу *K*. Для EOFB нешифрованный расширенный ключ передается в **ClearSalt** в рамках **paramS** (*sc*). Элемент **encryptedSaltingKey** должен передавать шифрованный медийный расширенный ключ, где функция шифрования должна применять сеансовый ключ ведущего объекта и исходное значение **paramSsaltIV** к медийному расширенному ключу *KS*. Для EOFB незашифрованный расширенный ключ (*ksc*) передается в **ClearSalt** в рамках **paramSsalt**. Элемент **clearSaltingKey** может содержать нешифрованный медийный расширенный ключ, и в этом случае **encryptedSaltingKey** должен остаться пустым, и наоборот. Передача нешифрованного расширенного ключа должна проводиться только в случае, если от этого не страдает безопасность данных, во всех остальных случаях рекомендуется, чтобы расширенный медийный ключ был зашифрован.

Ведущий объект должен быть подготовлен к приему зашифрованной защиты нового сеансового ключа медийной информации, посредством принятия команды **EncryptionUpdateCommand,** но должен продолжать использовать старый сеансовый ключ до получения **EncryptionUpdateAck**. Ведущий объект может применять новый сеансовый ключ, начиная с получения **encryptionUpdateAck**, в то время как ведомый объект может применять новый сеансовый ключ, начиная с получения **EncryptionUpdateCommand**.

ПРИМЕЧАНИЕ 1. – Ведущий объект может выбрать любое значение типа динамической полезной нагрузки для ведомого объекта, поскольку тип полезной нагрузки связан только с этим портом медиаканала.

ПРИМЕЧАНИЕ 2. – Для ведомого объекта нет необходимости явно подтверждать получение нового ключа. Ведущий объект способен сделать вывод о получении посланного ключа ведомым объектом, если получаемые медиаданные зашифрованы по новому типу полезной нагрузки.

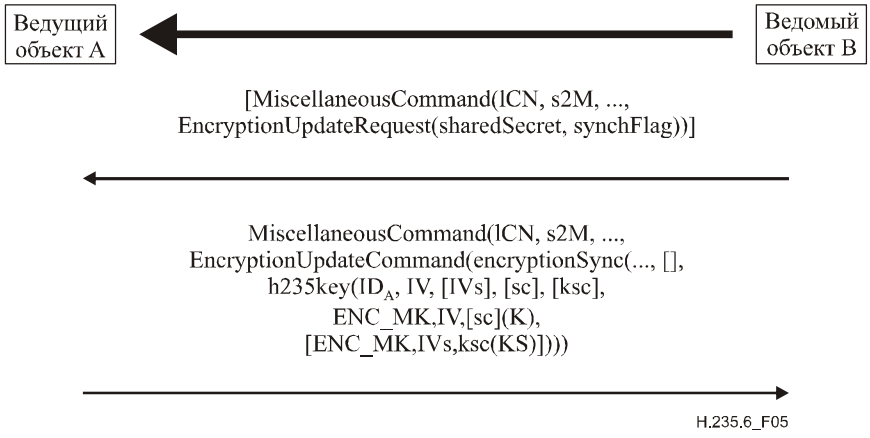


Рисунок 5/H.235.6 – Обновление сеансового ключа в логическом канале ведомого объекта

Рисунок 6 демонстрирует процедуры обновления ключа для логического канала, владельцем которого является ведущий объект. В случае если ведомый объект инициирует обновление ключа и запрашивает у ведущего объекта новый сеансовый ключ, ведомый объект должен послать ведущему объекту команду **MiscellaneousCommand**, где **logicalChannelNumber** должен содержать номер логического канала (его определяет ведущий), элемент **sharedSecret** должен быть установлен на истинное положение, флаг **direction** должен быть установлен на **masterToSlave**. Если, напротив,

ведущий объект инициирует обновление ключа, то это сообщение **EncryptionUpdateRequest** не должно посылаться.

Ведущий объект, или отвечая на запрос ведомого объекта, или от своего имени, должен давать команду **Encryption UpdateCommand,** где элемент **logicalChannelNumber** должен содержать номер логического канала, **direction** должен быть установлен на **masterToSlave**, элемент **encryptionSync** должен обеспечить **synchFlag** с номером новой динамической полезной нагрузки. Элемент **h235key** должен нести в себе новый сеансовый ключ. **h235key** должен содержать идентификатор ведущего объекта в **generalID** и примененный исходный вектор *IV* в **paramS**. Шифрованный медийный сеансовый ключ должен быть передан в рамках **encryptedSessionKey**, где функция шифрования должна применять ключ ведущего объекта и это значение в **paramS** к сеансовому ключу *K*. Для EOFB шифрованный расширенный ключ передается в **ClearSalt** через **paramS** (*sc*). Для EOFB **encryptedSaltingKey** должен передавать шифрованный медийный расширенный ключ, где функция шифрования должна применять сеансовый ключ ведущего объекта и исходное значение **paramSsaltIV** к расширенному медийному ключу *KS*. Для EOFB нешифрованный расширенный ключ (*ksc*) передается в **ClearSalt** в рамках **paramSsalt**. Элемент **clearSaltingKey** может содержать нешифрованный медийный расширенный ключ, и в этом случае **encryptedSalting Key** должен остаться пустым, и наоборот. Передача нешифрованного расширенного ключа может проводиться только в случае, если от этого не страдает безопасность данных, во всех остальных случаях рекомендуется, чтобы медийный расширенный ключ был зашифрован.

Ведомый объект должен подтвердить получение нового сеансового ключа отправлением команды **MiscellaneousCommand,** где элемент **logicalChannelNumber** должен содержать номер логического канала, а **encryptionUpdateAck** должен отражать номер новой динамической полезной нагрузки в **synchFlag**.

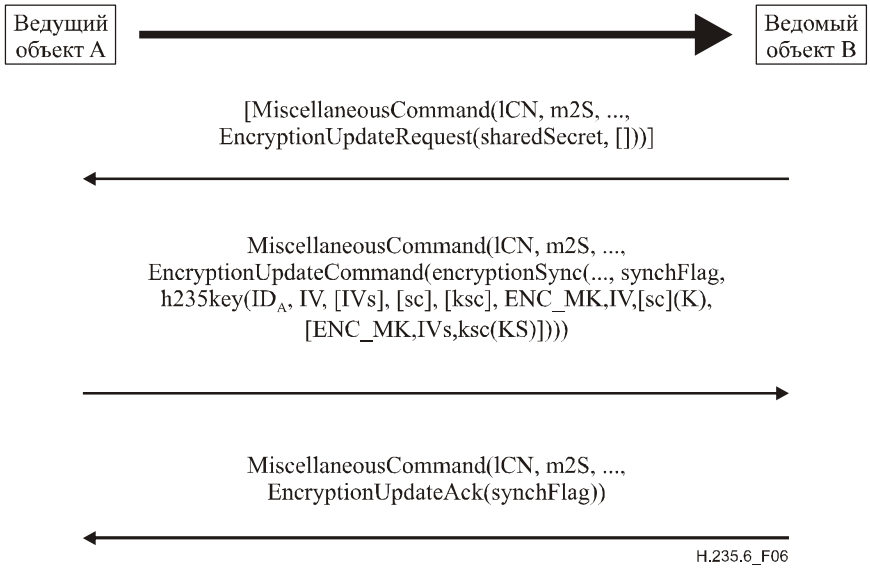


Рисунок 6/H.235.6 – Обновление сеансового ключа в логическом канале ведущего объекта

8.6.3 Обновление и синхронизация ключа, базирующегося на типе полезной нагрузки

Исходный ключ шифрования предоставляется ведущим объектом в совокупности с номером динамической полезной нагрузки в **synchFlag** (посредством **EncryptionSync** в Рек. МСЭ-Т H.245). Получатель(и) медиапотока должен(ны) приступать к началу использования этого ключа по получении данного номера полезной нагрузки в заголовке RTP.

Если оговоренный логический канал переносит только один тип полезной нагрузки, то величина **synchFlag** может заменить тип оговоренной полезной нагрузки в заголовке RTP. С другой стороны, если оговоренный логический канал может переносить несколько типов полезной нагрузки (даже если только в отдельных пакетах RTP), то пакеты RTP должны форматироваться согласно Стандарту в RFC 2198, со значением **synchFlag,** действующим в качестве инкапсулированного типа полезной

нагрузки), и фактического(их) типа(ов) полезной нагрузки находится в дополнительном(ых) блоке(ах) заголовка, как указано в Стандарте RFC 2198.

Новый(е) ключ(и) может (могут) быть распределен(ы) ведущей конечной точкой в любое время. Синхронизация более нового ключа с медиапотоком должна проявиться в изменении типа полезной нагрузки на новое динамическое значение.

ПРИМЕЧАНИЕ. – Конкретные значения интереса не представляют, поскольку они принимают новое значение с каждым новым распределением ключа.

8.7 Взаимодействия вне терминалов

8.7.1 Шлюз

Как установлено в 6.6/H.235.0, шлюз H.323 следует рассматривать как доверительный элемент. Это включает и шлюзы протоколов (H.323-H.320 и т. п.), и шлюзы защиты (прокси/брандмауэры). Секретность медиаданных между связывающимися друг с другом конечной точкой и шлюзовым устройством может быть обеспечена; но то, что происходит в дальнем конце шлюза следует рассматривать как незащищенный процесс по умолчанию.

8.7.2 Новые ключи

Процедуры, в общих чертах описанные в 8.5/H.323, дополнены MC, чтобы вывести участника из конференции. Ведущий объект может сформировать новые ключи шифрования для логических каналов (и не распределять их среди выведенных участников); это может использоваться для предотвращения просмотра выведенными участниками медиапотоков.

8.7.3 Доверительные элементы H.323

В общем случае MC(U), шлюзы и контроллеры доступа (если устанавливается модель, маршрутизируемая контроллером доступа) являются доверительными по отношению к конфиденциальности канала управления. Если канал установки соединений (H.225.0) защищен *и* маршрутизируется посредством контроллера доступа, он также должен быть доверительным. Если какой-либо из этих компонентов должен работать с медиапотоками (например, смешение, перекодирование), тогда, по определению, они должны также являться доверительными для секретности медиаданных.

Прокси/брандмауэры (хотя и не являющиеся элементами H.323) могут также быть доверительными, поскольку они разрывают соединения, и им также приходится управлять сообщениями и медиапотоками.

8.8 Многоточечные процедуры

8.8.1 Аутентификация

Аутентификация должна проводиться между конечной точкой и MC(U) таким же образом, как и при двусторонней конференции. MC(U) должен определять стратегию, касающуюся уровня и точности аутентификации. Как установлено в 6.6/H.235.0, MC(U) является доверительным элементом; существующие конечные точки во время конференции могут быть ограничены уровнем аутентификации, установленным MC(U). Новые команды **ConferenceRequest**/**ConferenceResponse** позволяют конечным точкам получать от MC(U) сертификаты других участников конференции. Как изложено в процедурах H.245, конечные точки в многосторонней конференции могут запрашивать сертификаты других конечных точек через MC, но не способны проводить прямую криптографическую аутентификацию в рамках канала H.245.

8.8.2 Секретность

MC(U) должен играть главную роль во всех обменах ведущий/ведомый и, по существу, должен обеспечить ключом(ами) шифрования участников многосторонней конференции. Секретность для отдельных источников при общей сессии (предполагая групповую передачу) может обеспечиваться индивидуальными или общими ключами. Два данных режима могут произвольно выбираться MC(U) и не должны контролироваться ни одной из конечных точек, за исключением режимов, разрешенных стратегией MC(U). Другими словами, общий ключ может использоваться для множества логических каналов, открытых из различных источников.

9 Процедуры шифрования медиапотока

Медиапотоки должны кодироваться с использованием алгоритма и ключа, как описывается в канале H.245. Рисунки 7 и 8 демонстрируют общий поток. Отметим, что заголовок транспорта прикрепляется к транспортному SDU после того, как SDU был зашифрован. Непрозрачные сегменты обозначают секретность. Как только новые ключи получены отправителем и использованы при шифровании, заголовок SDU должен каким-то образом указать получателю, что используется новый ключ. Например, в Рек. МСЭ-Т H.323, заголовок RTP (SDU) изменит свой тип полезной нагрузки, чтобы показать переключение на новый ключ.

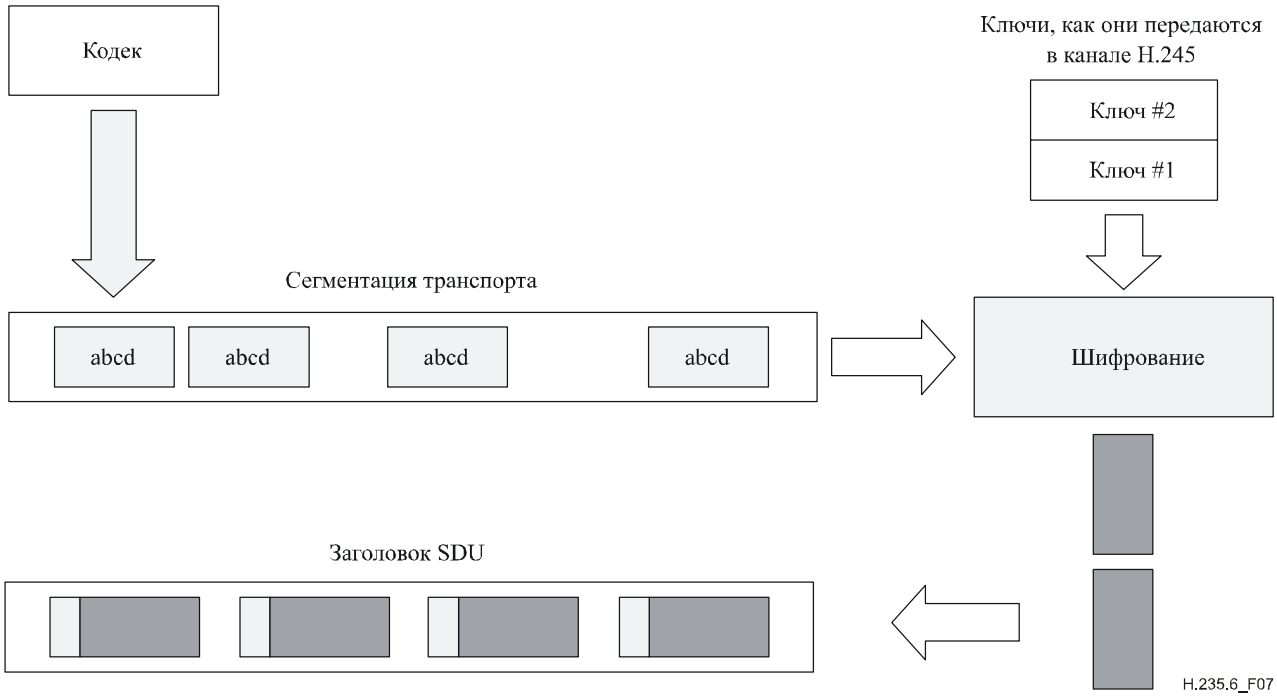


Рисунок 7/H.235.6 – Шифрование медиаданных

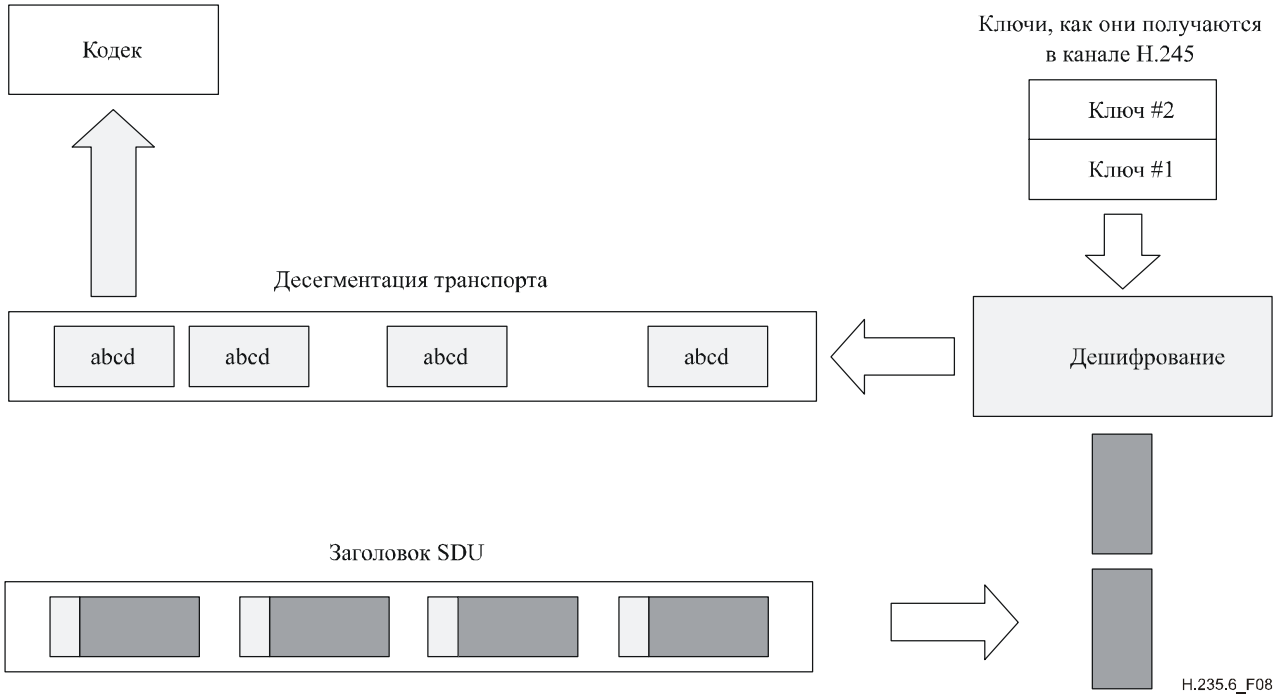


Рисунок 8/H.235.6 – Дешифрование медиаданных

9.1 Сеансовые ключи медиа

h235Key включен в **encryptionUpdate**. **h235Key** закодирован с помощью ASN.1 в контексте дерева ASN.1 H.235, и передан как непрозрачная цепочка октетов, относящаяся к H.245. Ключ может быть защищен с использованием одного из трех возможных механизмов, когда они передаются между двумя конечными точками.

- Если канал H.245 защищен, то к данным ключам не применяется дополнительная защита. Ключ передается "открытым текстом" относительно этого поля; используется выбор **secureChannel,** сделанный ASN.1.
- Если секретный ключ и алгоритм установлены за пределами канала H.245, взятого в целом (т. е. вне H.323 или в логическом канале **h235Control**), для шифрования данных ключа используется общий ключ; сюда включается результируемый зашифрованный ключ . В этом случае используется выбор **sharedSecret,** сделанный ASN.1.
- Сертификаты могут использоваться тогда, когда канал H.245 является защищенным, но также они могут использоваться дополнительно в случае защищенного канала H.245. Когда используются сертификаты, данные ключа шифруются с помощью публичного ключа сертификатов, конструкции **certProtectedKey** в ASN.

В любой точке в конференции получатель (или отправитель) может запросить новый ключ (**encryptionUpdateRequest**). Причина, по которой он может это сделать, заключается в предположении, что нарушена синхронизация одного из логических каналов. Ведущий терминал, получивший такой запрос, должен сформировать новый(ые) ключ(и) в ответ на эту команду. Ведущий терминал может также решить асинхронно распределять новый(ые) ключ(и), в этом случае он должен использовать сообщение **encryptionUpdate**.

По получении **encryptionUpdateRequest**, ведущий терминал должен послать **encryptionUpdate**. Если это многосторонняя конференция, MC (также ведущий) следует распределить новый ключ среди всех получателей до того, как он передаст этот ключ отправителю. Отправитель данных по логическому каналу должен использовать новый ключ как можно раньше после получения сообщения.

Отправитель (предполагается, что это не ведущий терминал) может также запросить новый ключ. Если отправитель – участник многосторонней конференции, должна быть выполнена следующая процедура:

- Отправитель должен передать MC (ведущий терминал) **encryptionUpdateRequest**.
- MC следует сформировать новый(ые) ключ(и) и передать сообщение **encryptionUpdate** всем участникам конференции, за исключением отправителя.
- После распределения новых ключей между всеми остальными участниками, MC должен передать отправителю **encryptionUpdate**. Тогда отправитель должен использовать новый ключ.

9.2 Защита от спама при передаче медийной информации

Получатель медиапотока RTP может изъявить желание бороться с "отказом от обслуживания" и потоком атак в обнаруженных портах RTP/UDP. Получатели, установив возможность защиты от спама, могут быстро определить, что полученный пакет RTP исходит из неправомочного источника и сбросить его.

Возможность защиты от спама, если она установлена, показывает использование механизма защиты от спама либо:

- для открытого текста медиаданных без шифрования (см. случай 1, ниже); либо
- в комбинации с зашифрованными медиаданными, когда **EncryptionCapability** указывает на алгоритм шифрования (см. случай 2, ниже).

Оба варианта обеспечивают упрощенную аудентификацию пакетов RTP в выбранных полях через рассчитанный код аутентификации сообщений (MAC). MAC может быть рассчитан с использованием идентификаторов объектов, определенных в 9.2.1. Криптографические алгоритмы основываются на:

- алгоритме шифрования (например, DES в режиме MAC см. ИСО/МЭК 9797-1 и 9797-2). DES-MAC отображается через OID "N", тогда как тройной DES-MAC отображается через OID "O"; или
- с использованием криптографической односторонней функции (например, SHA1). Используемый OID – "M".

Алгоритм MAC указывается в идентификаторе объекта **antiSpamAlgorithm**. Алгоритм OID неявно отображает также размер MAC; например, 1 блок = 64 битам для DES MAC. Чтобы сохранить ширину полосы пропускания, MAC может быть усечен, например, до 32-битового MAC, хотя при этом снижается уровень защиты; тогда он требует другого идентификатора объекта. Метод защиты от спама не зависит от какого-либо дополнительного шифрования полезной нагрузки (см. случаи 1 и 2, ниже).

Защита от спама использует следующий формат пакетов RTP (см. рисунок 9), где последовательность заполнения RTP интерпретируется следующим образом (см. пункт 5 в RFC 3550).

- Бит P в заголовке RTP должен быть установлен на 1.
- Байты заполнения должны быть добавлены к концу полезной нагрузки со следующим значением:

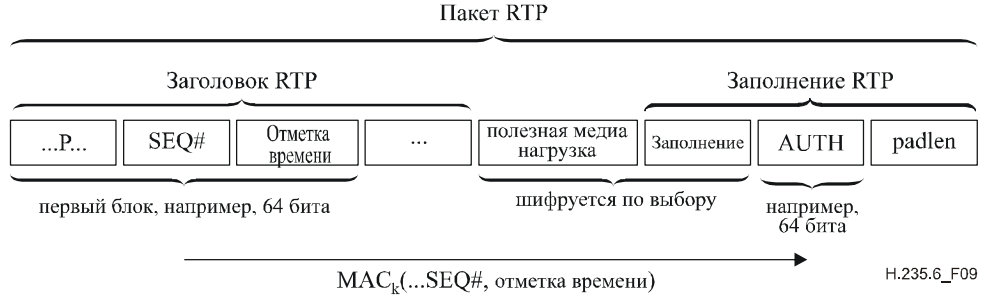


Рисунок 9/H.235.6 – Формат пакетов RTP для защиты от спама медийной информации

ПРИМЕЧАНИЕ 1. – Если защита от рассылки спама не используется, тогда не используются поля AUTH и padlen, и применяется обычный формат пакета RTP

1) *Случай только защиты от спама*

Этот случай применяется, когда медиаданные не зашифрованы и поля дополнения остаются пустыми. Последний октет дополнения RTP содержит счетчик, указывающий, сколько октетов в конце пакета RTP должно быть пропущено. Остальные байты заполнения передают MAC. MAC должен рассчитываться по первому криптоблоку заголовка RTP, включая переменную отметку времени и номер последовательности, используя согласованный алгоритм MAC **antiSpamAlgorithm** и применяя симметричный ключ. Статический или конфигурируемый "вручную" общий ключ, или динамически согласованный общий ключ *k* может использоваться согласно процедурам Рек. МСЭ-Т H.235.0. Для блоков большего размера (больше, чем 64 бита) должны быть взяты несколько дополнительных битов заголовка RTP, или даже первые биты полезной медианагрузки.

Для расчета MAC рекомендуется использовать ключ, полученный при распределении медиасеансовых ключей H.235; хотя применение сеансового ключа не используется для шифрования полезной нагрузки. Защищенное быстрое соединение с установлением ключа (см. Приложение J/H.323) или ручной ввод ключа может использоваться для управления ключами. Отправитель рассчитывает MAC, как было описано выше, и заносит результат в поле MAC внутри – поле AUTH заполнения RTP. Отправитель и получатель узнают размер поля AUTH и длину MAC из **antiSpamAlgorithm**.

Проверка MAC на стороне получателя должна быть сделана как можно скорее, если возможно уже со стеком RTP, в любом случае, не позднее, чем перед расшифровкой или декомпрессией полезной нагрузки. Получатель сначала пересчитывает MAC таким же образом, как это делал отправитель, и сравнивает рассчитанный MAC с полученным MAC в заполнении RTP. Если значения MAC не совпадают, заголовок RTP был изменен во время

передачи или был послан неправомочным объектом, который не обладает ключом. Таким образом, неаутентифицированный пакет RTP должен быть сброшен, а событие защиты может быть зарегистрировано; это указывает на возможную попытку атаки "отказа в обслуживании". В противном случае аутентифицированный пакет RTP может обрабатываться дальше, заполнение RTP удаляется, и полезная нагрузка пропускается через кодек.

ПРИМЕЧАНИЕ 2. – Упрощенный расчет/подтверждение MAC через шифрование DES включает в себя только одну операцию шифрования; в качестве альтернативы, SHA1 MAC рассчитывается на коротких частях пакетов фиксированной длинны, таким образом, криптооперации требуют на обработку абсолютно незначительные ресурсы.

2) *Случай защиты от спама и шифрования полезной нагрузки*

Этот случай применяется, когда одновременно шифруются медиаданные и запускается метод защиты от спама. Когда полезная нагрузка не совпадает с границами четных блоков, к ней перед MAC должно быть добавлено несколько битов заполнения. Шифрование полезной медианагрузки производится согласно данному пункту 9.

EncryptionCapability определяет алгоритм шифрования полезной нагрузки, в то время как **antiSpamAlgorithm** определяет метод защиты от спама. По соображениям безопасности, шифрование медиа и MAC должны использовать различные сеансовые ключи. Ключ MAC k рассчитывается пропусканием ключа шифрования K через одностороннюю хеш-функцию SHA1;

k = SHA1(K); из хешированного результата должно быть взято в сетевом порядке байтов достаточное число битов. Если **antiSpamAlgorithm** показывает алгоритм шифрования, из собранных битов должен быть сформирован правильный ключ шифрования, например, установка битов четности DES.

После того, как получатель успешно проверит аутентичность пакета RTP, полезная нагрузка расшифровывается и заполнение битами RTP затем сбрасывается. Общая процедура соответствует вышерассмотренному случаю 1.

9.2.1 Список идентификаторов объекта

В таблице 5 перечислены все OID, на которые делаются ссылки.

Таблица 5/H.235.6 – Идентификаторы объектов, используемые для защиты от спама

Опорное значение идентификатора объекта	Значение идентификатора объекта	Описание
"M"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 8}	Защита от спама с использованием HMAC-SHA1-96
"N"	{iso(1) identified-organization(3) oiw(14), secsig(3) algorithm(2) desMAC(10)}	Защита от спама с использованием DES (56 бит) MAC (см. ИСО/МЭК 9797-1 и 9797-2) с 64-битовым MAC
"O"	{iso(1) identified-organization(3) oiw(14) secsig(3) algorithm(2) desEDE(17)}	Защита от спама с использованием тройного DES (168 битов) MAC (см. ИСО/МЭК 9797-1 и 9797-2)

9.3 Вопросы, касающиеся RTP/RTCP

Использование шифрования потока RTP следует общей методологии, рекомендуемой в документе, указанном в [RTP]. Шифрование медиаданных должно проводиться попакетно, независимо от каждого пакета.

ПРИМЕЧАНИЕ. – Следует отметить, что если размер пакета RTP больше, чем размер пакета MTU, частичная потеря (фрагмента) сделает целый пакет RTP нешифруемым.

Заголовок RTP не должен шифроваться. Для аудио/видео кодеков, вся полезная нагрузка аудио/видео кодека, включая заголовок(и) полезной аудио/видео нагрузки должна шифроваться. Синхронизация новых ключей и зашифрованного текста основывается на динамическом типе полезной нагрузки (см. 8.6.3).

Предполагается, что шифрование применяется только к полезной нагрузке в каждом пакете RTP, заголовки RTP остаются "открытым текстом". Предполагается, что все пакеты RTP должны быть кратными целому числу октетов. К данной Рекомендации не относится капсулирование пакетов RTP на уровнях транспорта или сети. Все режимы должны признавать потерянные (или выбившиеся из последовательности) пакеты, в добавление к заполнению битами до подходящего кратного числа октетов.

Дешифрование потока должно проводиться без информации о состоянии, поскольку пакеты могут быть потерянными; каждый пакет следует дешифровать отдельно. Два требования режима блочного алгоритма должны действовать следующим образом:

9.3.1 Векторы инициализации

Большинство блочных режимов содержат некое "образование цепочки"; каждый цикл шифрования зависит некоторым образом от выходных результатов предыдущего цикла. Следовательно, в начале пакета, чтобы начать процесс шифрования, необходимо какое-то начальное значение блока (обычно называемое – вектором инициализации (IV)). Независимо от того, сколько октетов потока обрабатывается в каждом цикле шифрования, длина IV всегда равна длине блока. Все режимы, исключая режим электронная кодовая книга (ECB), требуют IV.

9.3.1.1 Вектор инициализации CBC

Вектор инициализации (IV) требуется при использовании блочного шифра в режиме CBC для шифрования полезных нагрузок пакета RTP. Размер IV такой же, как и размер блока в конкретном блочном шифре. Например, размер IV для DES и 3-DES – 64 бита, в то время как для AES он составляет 128 битов.

В случае CBC, IV должен быть составлен из первых *B* (где *B* – это размер блока) октетов: Seq#, сцепленных с Timestamp. Это формирует комбинацию битов *SSTTTT*, где *SS* – это 2-октетная RTP Seq# и *TTTT* – это 4-октетная отметка времени RTP. Эта комбинация должна повторяться до тех пор, пока не будут сформированы октеты *B*, усеченные как необходимо. Например, 64- и 128-битовые IV содержали бы *SSTTTTSS* и *SSTTTTSSTTTTSSTT*, соответственно. Следует отметить, что IV, сформированный таким образом, может создавать комбинацию ключа, которая считается "неустойчивой" для конкретного алгоритма.

9.3.1.2 Вектор инициализации EOFB

Однозначный исходный вектор IV для каждого пакета RTP в режиме EOFB должен рассчитываться следующим образом:

Каждый пакет RTP ассоциируется с явным индексом *i* 48-битового пакета, как определено в [SRTP], где $i = 2^{16} \times \text{ROC} + \text{SEQ}$, где SEQ – номер последовательности, взятый из заголовка RTP, а ROC – это 32-битовый счетчик с автоматическим переключением, считающий частоту конвертаций номера последовательности SEQ в 65535.

Первоначально счетчик с автоматическим переключением ROC должен быть установлен на ноль. Каждый раз, когда SEQ конвертируется по модулю 2^{16}, отправитель должен увеличить ROC на единицу по модулю 2^{32}.

Исходный вектор IV рассчитывается как ($i \parallel T$ [$\parallel i \parallel T \parallel$ ...]) с 48-битовым индексом *i* и 32-битовой отметкой времени *T,* взятыми из заголовка RTP, конкатенированного несколько раз, до тех пор, пока не заполнен размер блока. Символ || обозначает конкатенацию.

ПРИМЕЧАНИЕ. – Автоматический счетчик и IV хранятся и рассчитываются локально на каждой одноранговой стороне и не передаются.

Если получатель столкнулся с потерей или перестановкой пакетов, он должен рассчитать оцененный индекс *i* как:

$i = 2^{16} \times v + \text{SEQ}$, где *v* выбирается из ряда {ROC-1, ROC, ROC+1} по модулю 2^{32} таким образом, что *v* – ближайшее число (по смыслу 2^{48}) к величине $2^{16} \times \text{ROC} + s_l$, где s_l – это номер последовательности, хранящийся у получателя. После того, как пакет обработан, используя оцененный индекс, получатель должен решить надо ли обновлять s_l и ROC. Например, простым (но не к устойчивым ошибкам) методом является просто подставить s_l в SEQ (если $\text{SEQ} > s_l$) и, если значение $v = \text{ROC} + 1$ было

использовано, то необходимо обновить ROC на *v*; см. также [SRTP], пункт 3.2.1, для дальнейшей информации.

9.3.2 Заполнение

Режимы ECB и CBC всегда обрабатывают входные потоки по блоку, и поскольку CFB и OFB могут обрабатывать входной сигнал с любым числом октетов, N ($\leq B$), рекомендуется, чтобы $N = B$.

Для управления пакетами, чья полезная нагрузка не представляет собой множество блоков, существуют два метода:

1) Принудительный захват неполных блоков шифротекста для ECB и CBC; нет заполнения для CFB и EOFB.

2) Заполнение, предписанное [RTP], пункт 5.1.

[RTP], пункт 5.1 описывает метод заполнения, при котором полезная нагрузка должна добавляться ко множеству блоков. Последний октет должен быть установлен в соответствии с числом октетов заполнений (включая последний) и установкой *P* в заголовке RTP. Величина заполнения должна определяться обычным согласованием алгоритма шифрования.

Все реализации H.235 должны поддерживать обе схемы. Используемая схема может быть определена следующим образом: если в заголовке RTP установлен бит *P,* значит, пакет дополняется; если размер пакета не кратен *B* и бит *P* не установлен, тогда применяется метод Ciphertext Stealing, если пакет кратен B, а заполнение не применяется.

9.3.3 Защита RTCP

Применение криптографических методов к элементам RTCP – материал для дальнейших исследований.

9.3.4 Защищенный поток полезной нагрузки

Сети, основанные на H.323, при использовании, например, для модемного соединения через IP, применяют сигнализацию H.245 для установки и согласования речевого канала данных и RTP для формирования пакетов группового потока полезной нагрузки (MPS).

При одиночном медиапотоке с одним типом полезной нагрузки или FEC для другого канала, тип динамической полезной нагрузки в элементе **encryptionSync** должен заменить тип полезной нагрузки по умолчанию.

Для инкапсулированных потоков (т. е. с избыточным кодированием или RFC 2198 кодированных FEC), тип динамической полезной нагрузки в рамках **encryptionSync** должен заменить инкапсулированный тип полезной нагрузки.

Для групповых потоков полезной нагрузки, тип динамической полезной нагрузки в **syncflag** элемента **encryptionSync** не должен приниматься во внимание, а вместо этого должны использоваться (необязательные) типы полезной нагрузки в рамках **multiplePayloadStreamElement**.

EncryptionUpdateCommand должна использоваться для процедуры обновления улучшенного ключа для распространения данных нового сеансового ключа (см. 8.6.2). **multiplePayloadStream** используется только тогда, когда групповой поток полезной нагрузки должен быть перенастроен по ключу, и в этом случае тип динамической полезной нагрузки в рамках **EncryptionSync** во внимание не принимается.

9.3.5 Взаимодействие с Рекомендацией J.170

Подлежит дальнейшему изучению.

9.4 Тройной DES во внешнем режиме CBC

168-битовый тройной DES во внешнем режиме CBC, как показано на рисунке 10, *следует* использовать в пределах этого профиля защиты. На рисунке, каждый k_i указывает на 56-битовый ключ. Разные 56-битовые ключи *должны* быть использованы в каждом блоке шифрования (E) и дешифрования (D). Неизвестно случаев, чтобы какой-либо из 64 неустойчивых ключей для DES вызвал какую-либо неустойчивость в тройном DES. Однако реализации, соответствующие

требованиям этого профиля, должны отклонять ключ, если в него включен неустойчивый ключ DES (см. RFC 2405).

Дополнительную информацию о тройном DES можно получить в [Schneier] и [RFC2405].

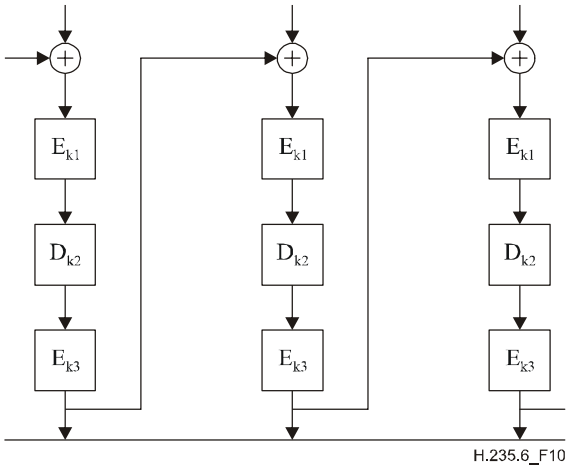


Рисунок 10/H.235.6 – Шифрование в тройном DES во внешнем режиме CBC

9.5 Алгоритм DES, действующий в режиме EOFB

Речевые сообщения могут шифроваться с использованием алгоритма DES, работающего в режиме поточного шифрования сцеплений блоков в EOFB. Режим EOFB позволяет использовать в реализациях параллелизм. При работе в режиме EOFB, рекомендуется по соображениям, как производительности, так и безопасности, возвратить к источнику весь криптоблок (т. е. полный 64-битовый, например, для DES с $n = j = 64$). Однако из-за того, что EOFB не обеспечивает сцепление блоков и битов, EOFB может подвергаться атакам, в зависимости от статистических свойств данных входного нешифрованного текста. Таким образом, обновление ключей (см. 8.6) должно производиться регулярно, но не позднее чем до свертывания исходных значений. Для расчета исходного значения см. 9.3.1.2.

9.6 Тройной DES во внешнем режиме EOFB

168-битовый тройной DES во внешнем режиме EOFB, как показано на рисунке 11, может использоваться в данном профиле защиты. На рисунке каждое k_i указывает на 56-битовый ключ. Отличный от него 56-битовый ключ *должен* быть использован в обоих блоках шифрования (E) и дешифрования (D). Неизвестно случаев, чтобы какой-либо из 64-битовых неустойчивых ключей для DES вызвал какую-либо неустойчивость в рамах тройного DES. Однако реализации, соответствующие требованиям этого профиля, должны отклонять ключ, если в него включен неустойчивый ключ DES (см. RFC 2405).

Дополнительную информацию о тройном DES можно получить в [Schneier] и [RFC2405].

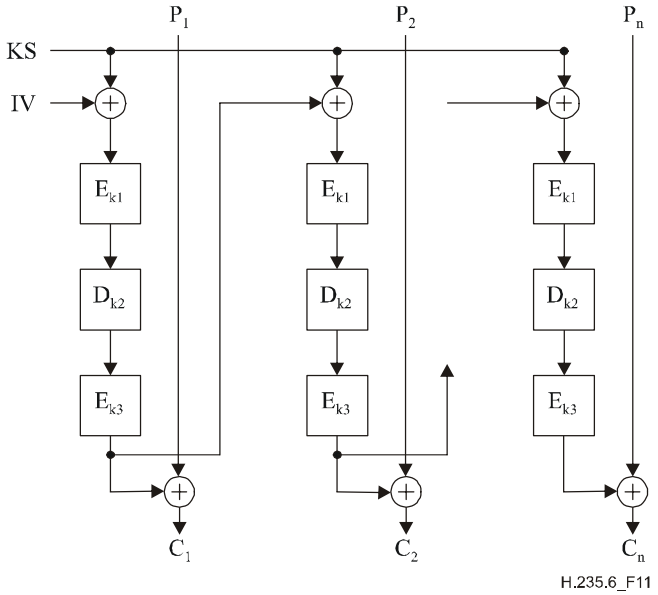


Рисунок 11/H.235.6 – Шифрование тройного DES во внешнем режиме EOFB

10 Санкционированный перехват

Подлежит дальнейшему изучению (см. [LI]).

11 Перечень идентификаторов объектов

Таблица 6 содержит все упомянутые OIDs (см. также [OIW] и [WEBOIDs]). Данные идентификаторы объектов для H.235v1 (Рек. МСЭ-Т H.235v1) и для H.235v2 (Рек. МСЭ-Т H.235v2).

Таблица 6/H.235.6 – Идентификаторы объектов

Обозначение идентфикатора объекта	Значение идентификатора объектов	Описание
"DHdummy"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 40} {itu-t (0) recommendation (0) h (8) 235 version (0) 3 40}	Нестандартная DH группа, представленная явно
"DH1024"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 43} {itu-t (0) recommendation (0) h (8) 235 version (0) 3 43}	1024-битовая DH группа
"DH1536"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 44}	1536-битовая DH группа
"X"	{iso(1) member-body(2) us(840) rsadsi(113549) encryptionalgorithm(3) 2}	Шифрование речевых сообщений с использованием RC2-совместимого (56 битов) или RC2- совместимого в режиме CBC и 512-битовой DH группы.
"X1"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 27}	Шифрование речевых сообщений с использованием RC2-совместимого (56 битов) или RC2-совместимого в режиме EOFB и 512-битовой DH группы.

Таблица 6/H.235.6 – Идентификаторы объектов

Обозначение идентификатора объекта	Значение идентификатора объектов	Описание
"Y"	{iso(1) identified-organization(3) oiw(14) secsig(3) algorithm(2) descbc(7)}	Шифрование речевых сообщений с использованием DES (56 битов) в режиме CBC и 512-битовой DH группы.
"Y1"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 28}	Шифрование речевых сообщений с использованием DES (56 битов) в режиме EOFB и 512-битовой DH группы с 64-битовой обратной связью
"Z1"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 29}	Шифрование речевых сообщений с использованием тройного DES (168 битов) во внешнем режиме EOFB и 1024-битовой DH группы с 64-битовой обратной связью
"Z2"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 30}	Шифрование речевых сообщений с использованием AES (128 битов) в режиме EOFB и 1024-битовой DH группы
"Z3"	{joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) 3 nistAlgorithm(4) aes(1) cbc(2)}	Шифрование речевых сообщений с использованием AES (128 битов) в режиме CBC и 1024-битовой DH группы
"Z"	{iso(1) identified-organization(3) oiw(14) secsig(3) algorithm(2) desEDE(17)}	Шифрование речевых сообщений с использованием тройного DES (168 битов) во внешнем режиме CBC и 1024-битовой DH группы.

Дополнение I

Подробное описание реализации H.323

I.1 Метод заполнения зашифрованного текста

Описание процесса принудительного "захвата" зашифрованного текста представлено на страницах 191 и 196 [Schneier]. Рисунки с I.1 по I.5 иллюстрируют сам метод.

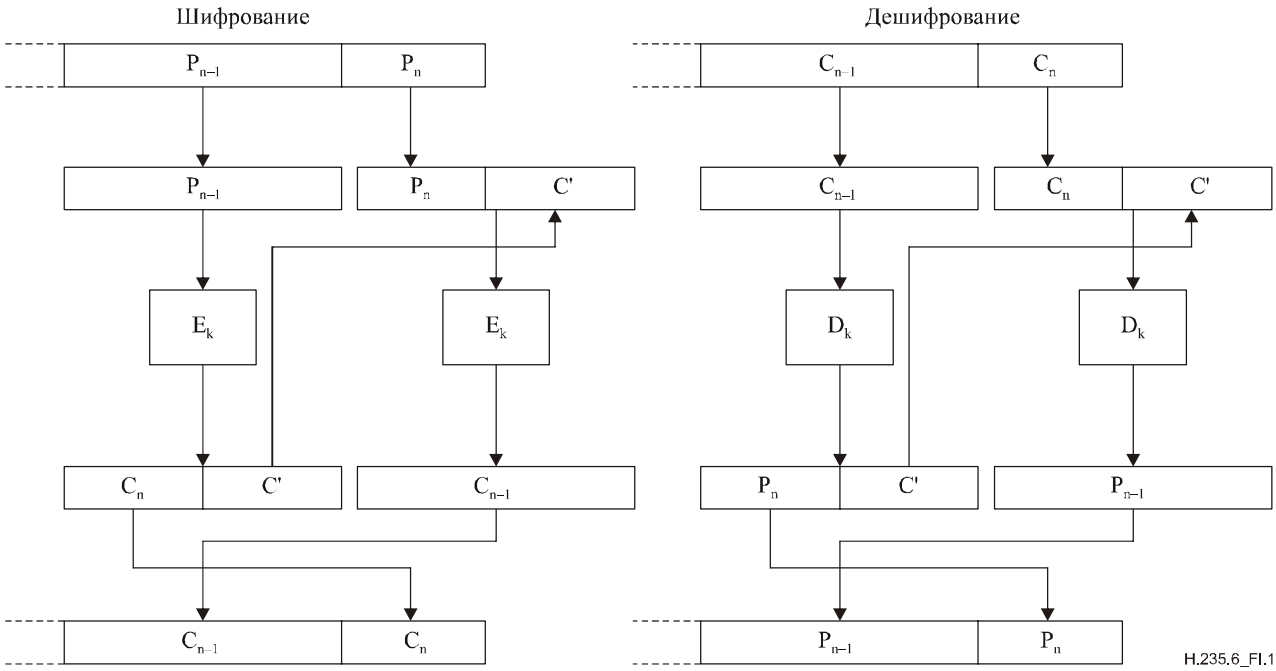


Рисунок I.1/H.235.6 – Принудительный "захват" зашифрованного текста в режиме ECB

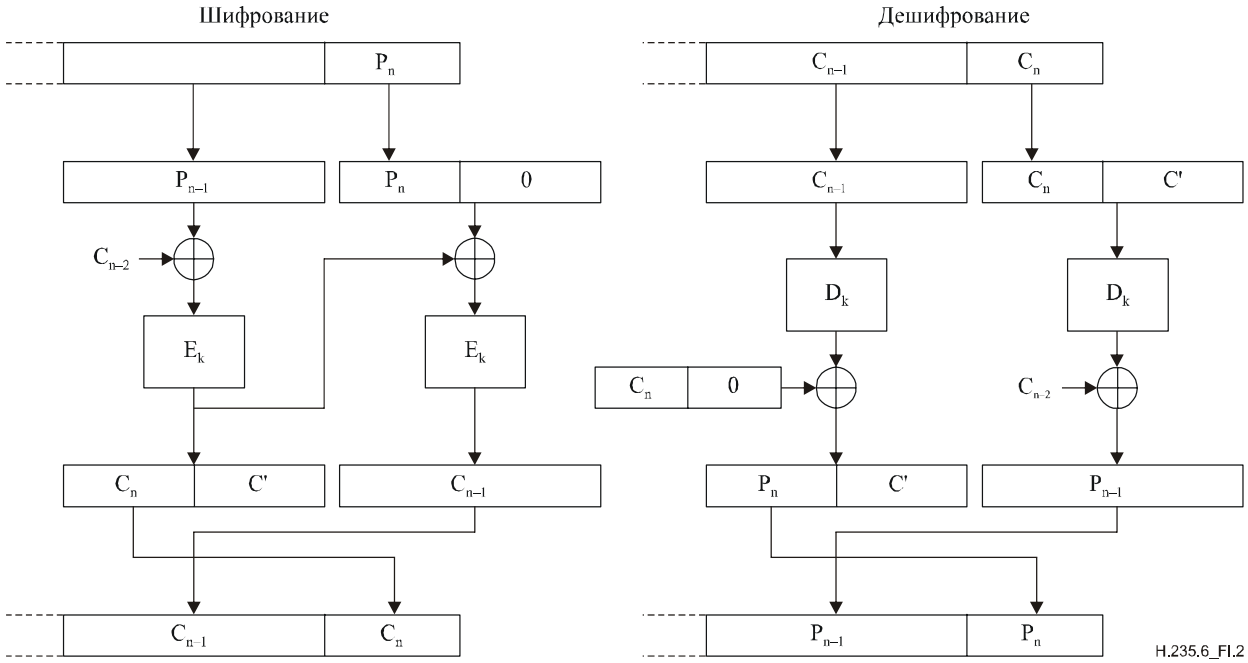


Рисунок I.2/H.235.6 – Принудительный "захват" зашифрованного текста в режиме CBC

ПРИМЕЧАНИЕ. – Принудительный "захват" зашифрованного текста в режимах ECB или CBC требует, чтобы полезная нагрузка включала, по крайней мере, один полный блок. Реализации, использующие принудительный "захват" зашифрованного текста в режиме ECB или CBC, должны удостовериться, что полезная нагрузка всегда содержит не менее одного криптоблока; например, при надлежащем выборе частоты выборки/пакетирования или алгоритма шифрования.

В случае если полезная нагрузка охватывает менее одного отдельного блока, исходный вектор (IV) должен использоваться в качестве предыдущего блока зашифрованного текста, при применении режима принудительного "захвата" зашифрованного текста в режиме CBC.

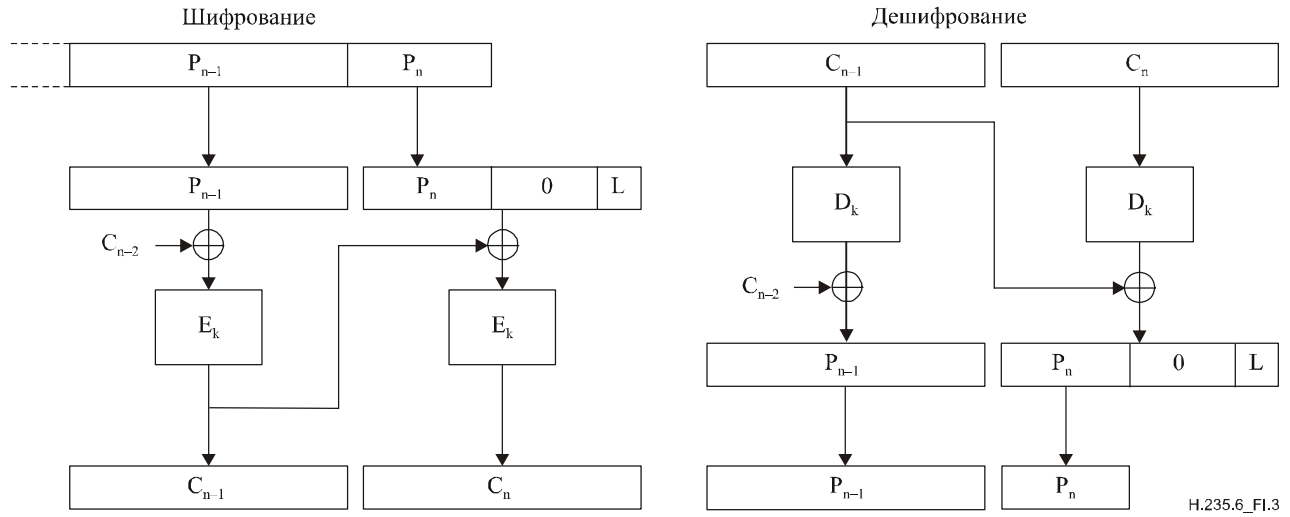


Рисунок I.3/H.235.6 – Заполнение нулями в режиме CBC

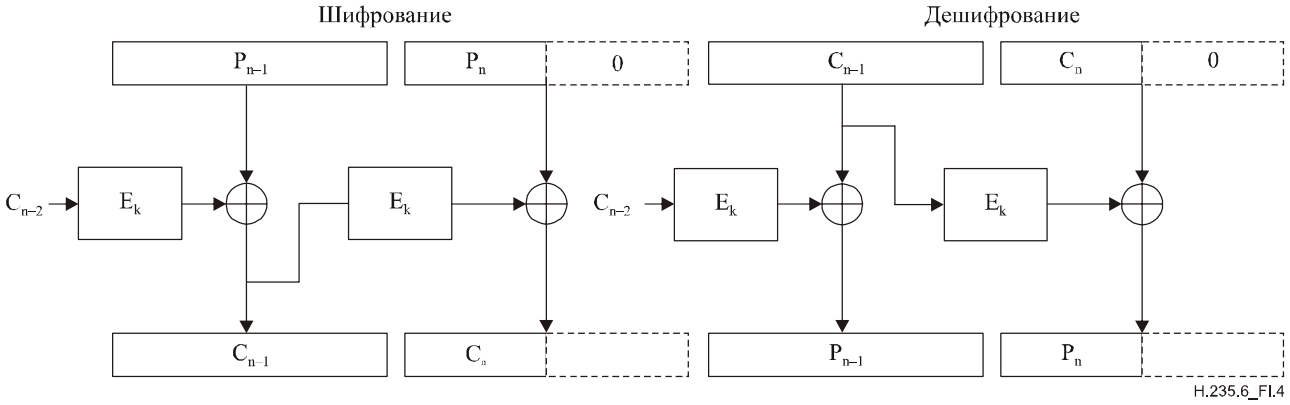
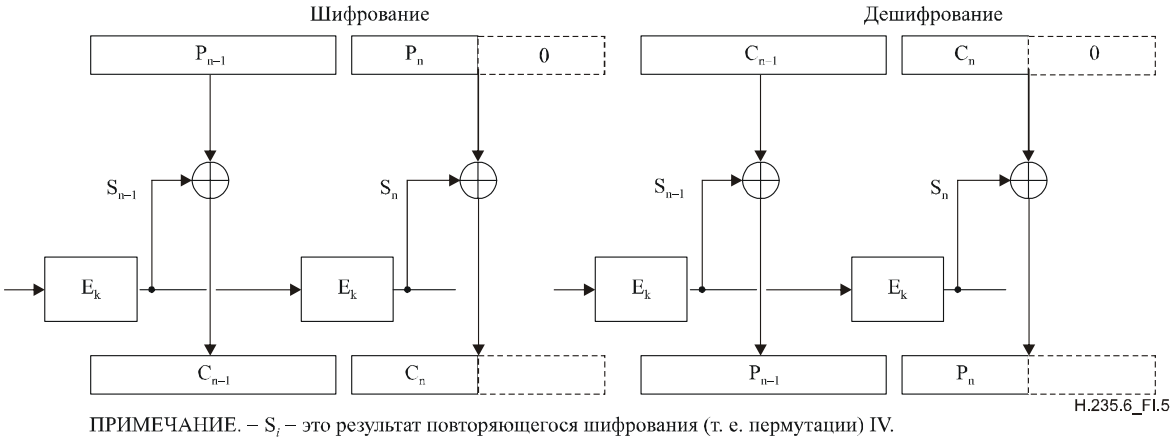


Рисунок I.4/H.235.6 – Заполнение нулями в режиме CFB



ПРИМЕЧАНИЕ. – S_i – это результат повторяющегося шифрования (т. е. пермутации) IV.

Рисунок I.5/H.235.6 – Заполнение нулями в режиме OFB

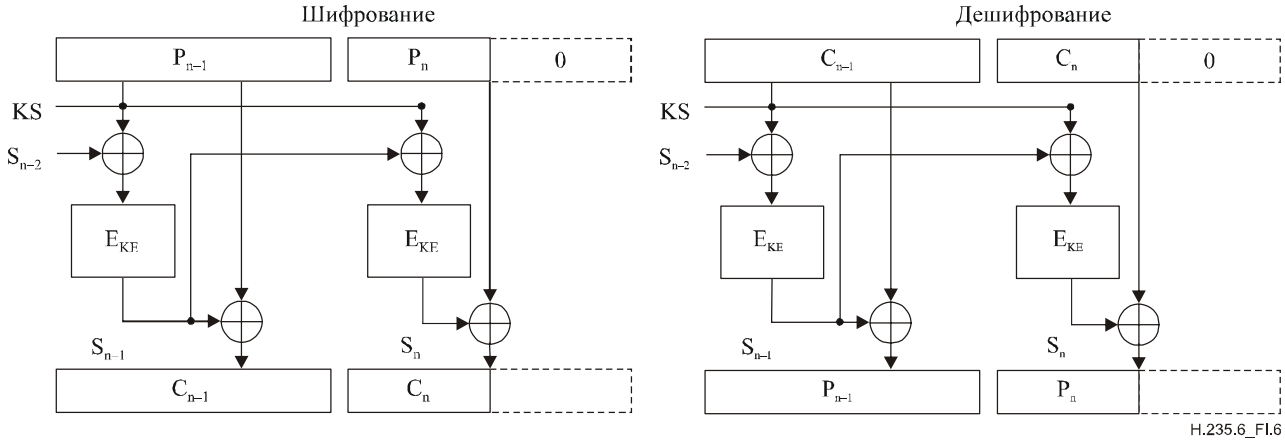


Рисунок I.6/H.235.6 – Заполнение нулями в режиме EOFB

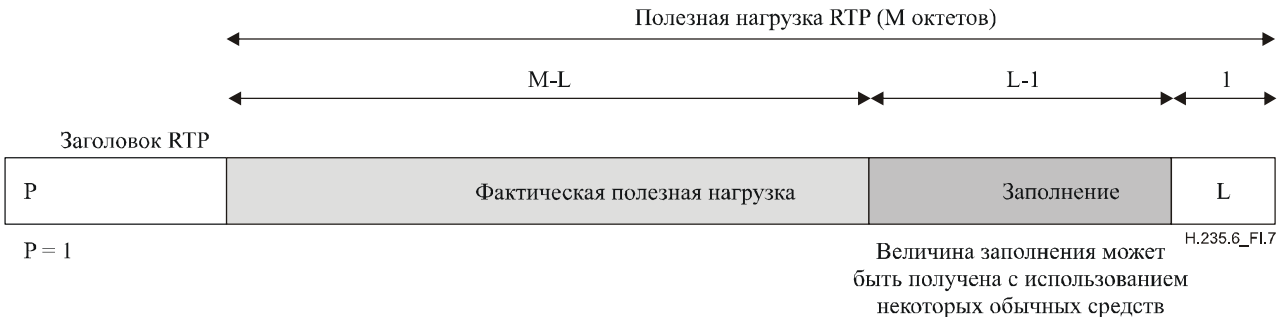


Рисунок I.7/H.235.6 – Заполнение, как предписано RTP

I.2 Новые ключи

Процедуры, описанные в 8.5/H.323, завершаются МС выводом какого-либо участника из конференции. Ведущий терминал может создавать новые ключи шифрования для логических каналов (и не передавать их выведенной стороне); это может использоваться для того, чтобы воспрепятствовать мониторингу медиапотоков выведенной стороной.

СЕРИИ РЕКОМЕНДАЦИЙ МСЭ-Т

Серия A	Организация работы МСЭ-Т
Серия D	Общие принципы тарификации
Серия E	Общая эксплуатация сети, телефонная служба, функционирование служб и человеческие факторы
Серия F	Нетелефонные службы электросвязи
Серия G	Системы и среда передачи, цифровые системы и сети
Серия H	**Аудиовизуальные и мультимедийные системы**
Серия I	Цифровая сеть с интеграцией служб
Серия J	Кабельные сети и передача сигналов телевизионных и звуковых программ и других мультимедийных сигналов
Серия K	Защита от помех
Серия L	Конструкция, прокладка и защита кабелей и других элементов линейно-кабельных сооружений
Серия M	Управление электросвязью, включая СУЭ и техническое обслуживание сетей
Серия N	Техническое обслуживание: международные каналы передачи звуковых и телевизионных программ
Серия O	Требования к измерительной аппаратуре
Серия P	Качество телефонной передачи, телефонные установки, сети местных линий
Серия Q	Коммутация и сигнализация
Серия R	Телеграфная передача
Серия S	Оконечное оборудование для телеграфных служб
Серия T	Оконечное оборудование для телематических служб
Серия U	Телеграфная коммутация
Серия V	Передача данных по телефонной сети
Серия X	Сети передачи данных, взаимосвязь открытых систем и безопасность
Серия Y	Глобальная информационная инфраструктура, аспекты межсетевого протокола и сети последующих поколений
Серия Z	Языки и общие аспекты программного обеспечения для систем электросвязи

Отпечатано в Швейцарии
Женева, 2007 г.