УДК 004.056

ИССЛЕДОВАНИЕ ВЕРОЯТНОСТНО-ВРЕМЕННЫХ ХАРАКТЕРИСТИК ПРОТОКОЛА РАСПРЕДЕЛЕНИЯ КЛЮЧЕЙ ЗАЩИЩЕННОЙ IP-ТЕЛЕФОНИИ

М. М. Ковцур,
аспирант
В. Н. Никитин,
канд. техн. наук, доцент
Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича
А. В. Винель,
канд. техн. наук, ведущий научный сотрудник
ЗАО «НПФ ИНСЕТ», г. Москва

Разработана математическая модель криптографического протокола распределения ключей IP-телефонии Zimmermann Real-time Transport Protocol в виде вероятностного графа. Представлены теоретические зависимости вероятностно-временных характеристик данного протокола от параметров канала связи: задержки пакетов и вероятности битовых ошибок. Выполнено сравнение полученных теоретических оценок с результатами экспериментального моделирования.

Ключевые слова *— ключ, криптографический протокол, канал с ошибками, среднее время выполнения, вероятность успешного завершения, ZRTP, IP-телефония.*

Введение

В отличие от традиционной телефонии, использующей коммутацию каналов (аналоговых или цифровых), IP-телефония — это технология, обеспечивающая передачу речевого сигнала с применением коммутации пакетов в IP-сетях. В IP-телефонии, как правило, применяются протокол Real-time Transport Protocol (RTP)/Real-time Transport Control Protocol (RTCP) [1] для передачи голоса и один из протоколов сигнализации Session Initiation Protocol (SIP) [2], H.323, Media Gateway Control Protocol (MGCP) или H.248 для установления и поддержания соединения.

Наибольшее распространение в настоящий момент получил протокол SIP, отличающийся простотой реализации, гибкостью и расширяемостью.

При вызове вначале отрабатывает протокол SIP, позволяющий установить соединение между корреспондентами. Как только корреспондент снимает трубку, начинает работу протокол RTP/RTCP. Сценарии установления соединений представлены на рис. 1, а схема обмена сообщениями — на рис. 2, *а*.

В силу общедоступности используемых каналов передачи голосовой информации особую актуальность приобретает обеспечение конфиденциальности VoIP. Для этих целей разработаны протоколы обеспечения безопасности IP-телефонии, которые можно разделить на 3 категории:

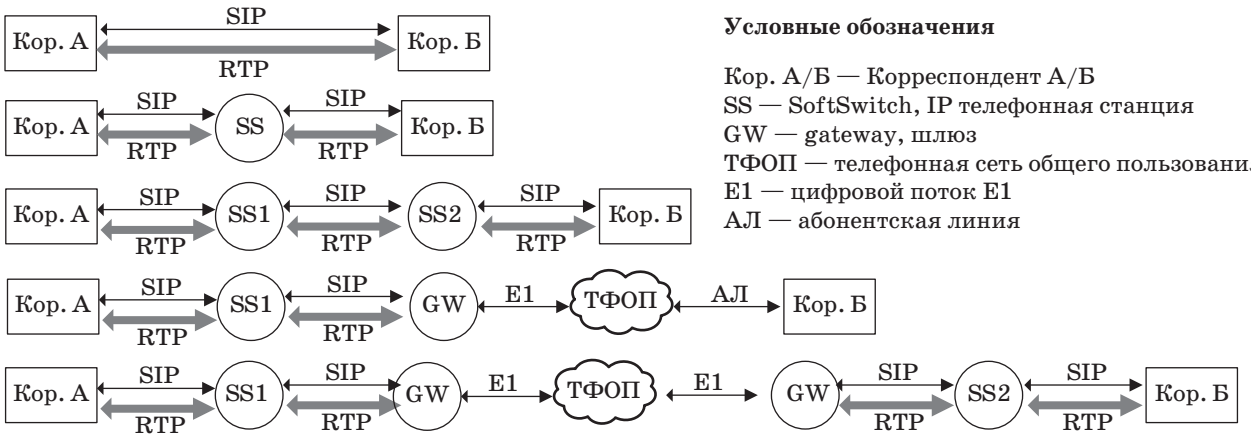
— протоколы защиты сигнализации (Secured SIP);

— протоколы защиты медиаинформации (SRTP);

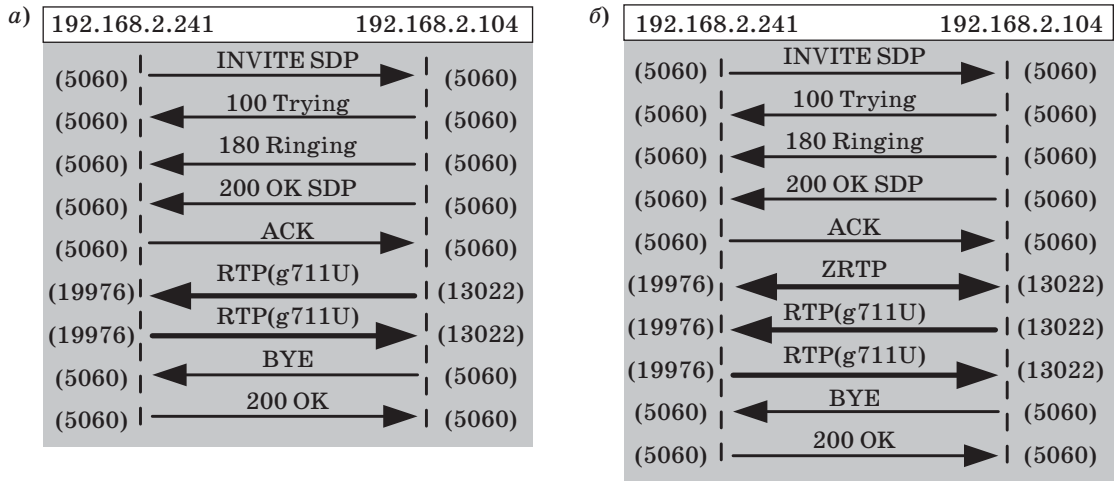
— протоколы генерации/распределения ключей для протоколов защиты медиаинформации (Multimedia Internet KEYing (MIKEY), Session Description Protocol Security (SDES), ZRTP, Datagram Transport Layer Security (DTLS)).

Протокол Secured SIP работает по аналогии с протоколом HyperText Transfer Protocol Secure (HTTPS), когда между корреспондентом и сервером организовывается туннель с использованием сертификатов и открытого ключа (Secure Sockets Layer — SSL). Все SIP-сообщения (сигнализация) передаются по этому туннелю.

Для обеспечения безопасности передачи речи широко используется защищенный протокол ре-



■ ***Рис. 1.*** Типовые сценарии соединений в IP-телефонии



■ ***Рис. 2.*** Схема обмена сообщениями при соединении двух корреспондентов с использованием протоколов SIP/RTP (*а*) и SIP/ZRTP/SRTP (*б*)

ального времени SRTP [3], который выполняет функции криптографической защиты (шифрование и аутентификацию сообщений) совместно с одним из протоколов, реализующих генерацию и распределение ключей.

Проблемы безопасности протокола ZRTP и его прототипа (алгоритма Диффи — Хелмана (Diffie-Hellman, DH)) исследованы в работах [4, 5]. Однако влияние его на характеристики своевременности предоставления связи не изучалось. Данная работа содержит результаты исследования вероятностно-временных характеристик одного из наиболее перспективных протоколов генерации ключей — протокола ZRTP [6, 7].

Поскольку протокол ZRTP работает сразу после завершения работы протокола сигнализации, время установления защищенного соединения увеличивается на величину времени выполнения протокола ZRTP. Пример установления соединения в защищенном режиме представлен на рис. 2, *б*.

При работе по идеальному каналу связи протоколы SIP/ZRTP будут иметь фиксированное время выполнения, равное сумме времен, необходимых для каждой итерации протокола, но при наличии задержек и ошибок в канале связи время выполнения будет случайной величиной. Поскольку время установления соединения в телефонной связи является нормированной величиной, задача оценки вероятностно-временных характеристик протокола ZRTP по каналам с задержками и ошибками является актуальной задачей, которая решается в данной статье.

Параметры, определяющие вероятностно-временные характеристики протокола ZRTP

Протокол ZRTP реализует функции генерации ключевых параметров SRTP сессии, аутентификации корреспондентов, обеспечения конфи-

денциальности обмена сообщениями протокола, защиты от атаки вторжения посередине (Man In The Middle — MITM). Эти задачи решаются последовательно (рис. 3).

Особенностью протокола является передача сообщений протокола внутри RTP-пакетов при сохранении их совместимости с RTP\AVP (Audio and Video Payload) профилем. В этом случае ZRTP-

	Корреспондент А			Корреспондент Б	
	Вычисления	Прием/передача		Прием/передача	Вычисления
Фаза 1		Hello A	Hello →	Hello A	Версия ZRTP, опции, ZID корреспондента А
	Получение сообщения как правило означает, что партнер поддерживает ZRTP	HelloACK	← HelloACK	HelloACK	
	Версия ZRTP, опции, ZID корреспондента Б	Hello Б	← Hello	Hello Б	
		HelloACK	HelloACK →	HelloACK	
Фаза 2		Согласование ролей корреспондентов и выполнение алгоритма Диффи—Хелмана: корреспондент Б — инициатор, корреспондент А — респондент			
	svr — случайное число $pvr = g^{svr} \bmod p$	ZID корреспондента Б, опции, *hvi*	← Commit	ZID корреспондента Б, опции, *hvi*	*svi* — случайное число $pvi = g^{svi} \bmod p$ *rs1IDi* = HMAC(*rs1*, «Initiator») *rs2IDi* = HMAC(*rs2*, «Initiator») *auxsecretIDi* = HMAC(*auxsecret, Initiator's H3*) *pbxsecretIDi* = HMAC(*pbxsecret*, «Initiator») *DHPart2* = *pvi* + *rs1IDi* + *rs2IDi* *hvi* = hash(DHPart2‖Hello корреспондента А)
	rs1IDr = HMAC (*rs1*, «Responder») *rs2IDr* = HMAC (*rs2*, «Responder») *auxsecretIDr* = HMAC (*auxsecret*, *Responder's H3*) *pbxsecretIDr* = HMAC (*pbxsecret*, «Responder»)	*pvr, rs1IDr, rs2IDr, auxsecretIDr, pbxsecretIDr*	DHPart1 →	*pvr, rs1IDr, rs2IDr, auxsecretIDr, pbxsecretIDr*	$DHR = pvr^{svi} \bmod p$
Фаза 3	*total_hash* = hash (Hello корреспондента А ‖ Commit ‖ DHPart1 ‖ DHPart2) *s0* = *hash(c‖DHR‖ «ZRTP-HMAC-KDF» ‖ZIDi‖ IDr‖total_hash‖len(s1)‖ s1‖len(s2)‖s2‖len(s3)‖s3)*	*pvi, rs1IDi, rs2IDi, auxsecretIDi, pbxsecretIDi*	← DHPart2	*pvi, rs1IDi, rs2IDi, auxsecretIDi, pbxsecretIDi*	*total_hash* = hash (Hello корреспондента А ‖ Commit ‖ DHPart1 ‖ DHPart2) *s0* = *hash(c‖DHR‖ «ZRTP-HMAC-KDF» ‖ZIDi‖ ZIDr‖total_hash‖len(s1)‖s1‖len(s2)‖s2‖ len(s3)‖s3)*
		Генерация ключей ZRTP и главных ключей SRTP			
	rs0 = HMAC(*s0*, «retained secret») *zrtpkeyr* = KDF(*s0*, «Responder ZRTP key», KDF_Context) *zrtpkeyi* = KDF(*s0*, «Initiator ZRTP key», KDF_Context)	AES_CFB(*MAC, D, A, V, E flags, sig*)(*zrtpkeyr*) CFB инициализирующий вектор	Confirm1 →	AES_CFB(*MAC, D, A, V, E flags, sig*)(*zrtpkeyr*)	*rs0* = HMAC(*s0*, «retained secret») *zrtpkeyi* = KDF(*s0*, «Initiator ZRTP key», KDF_Context) *zrtpkeyr* = KDF(*s0*, «Responder ZRTP key», KDF_Context)
Фаза 4		AES_CFB(*MAC, D, A, V, E flags, sig*)(*zrtpkeyi*)	← Confirm2	AES_CFB(*MAC, D, A, V, E flags, sig*)(*zrtpkeyi*) CFB инициализирующий вектор	
		Conf2ACK	Conf2ACK →	Conf2ACK	

■ ***Рис. 3.*** Диаграмма взаимодействия корреспондентов при выполнении протокола ZRTP

несовместимым устройством ZRTP-пакеты просто отклоняются и не влияют на установленное соединение, которое будет продолжено в незащищенном режиме. Параметры сообщений представлены в табл. 1 [7].

Протокол требует операции определения сторон — инициатора и отвечающего (респондента), поэтому она выполняется на первой фазе протокола, когда корреспонденты обмениваются сообщениями Hello. Эти сообщения содержат информацию о поддерживаемых партнерами протоколах для определения возможности использовать SRTP: поддерживаемых алгоритмах хеширования, алгоритмах шифрования, типах аутентификационных тегов, протоколах согласования ключей и др. Предусмотрена повторная передача сообщения Hello до 20 раз, после чего принимается решение о невозможности продолжать выполнение протокола ZRTP и устанавливать сессию в защищенном режиме. Повторная передача данного сообщения производится с задержкой переменной величины 50, 100, 200 мс, причем, начиная с четвертой передачи, задержка имеет постоянную величину 200 мс. Каждое принятое сообщение Hello подтверждается ответным сообщением HelloACK. Для перехода протокола в следующую фазу оба корреспондента должны получить сообщение Hello и хотя бы один из корреспондентов должен получить сообщение HelloACK. Корреспондент, получивший сообщение HelloACK, принимает на себя роль инициатора сессии.

■ ***Таблица 1***. Параметры сообщений протокола ZRTP

Назначение сообщения	Обозначение	Полная длина UDP-пакета, бит	Подтверждение	Число повторных передач (max)
Согласование параметров и возможностей корреспондентов	Hello	1392	Нужно	20
Подтверждение получения Hello-сообщения	HelloACK	650	Нет	
Согласования хеш-функций hvi	Commit	1392	Нужно	10
Первое сообщение обмена ключами Диффи — Хелмана	DHPart1	4208	Нет	
Второе сообщение обмена ключами Диффи — Хелмана	DHPart2	4208	Нужно	10
Подтверждение обмена с применением сгенерированного общего ключа	Confirm1	1072	Нет	
Подтверждение обмена с применением сгенерированного общего ключа	Confirm2	1072	Нужно	10
Подтверждение Confirm2	Conf2ACK	560	Нет	

Перед началом второй фазы каждый из корреспондентов генерирует свое случайное число и производит вычисление DH:

$$\text{первый корреспондент: } pvi = g^{svi} \bmod(p);$$

$$\text{второй корреспондент: } pvr = g^{svr} \bmod(p),$$

где *svi* и *svr* — случайные числа, закрытые ключи инициатора и респондента для алгоритма Диффи — Хелмана.

После этого корреспонденты подготавливают сообщение DHPart1/DHPart2 и формируют параметр *hvi* как укороченную до 256 бит хеш-функцию от конкатенации DHPart1/DHPart2 и сообщения Hello корреспондента:

$$hvi = hash(\text{DHPart1 or DHPart2} \| \text{Hello}).$$

Параметр *hvi* предназначен для проверки правильности аутентификации и подтверждения выбора инициатора. Он передается в составе сообщения Commit.

Инициатор первым посылает сообщение Commit. В том случае если оба устройства выбирают роль инициатора и одновременно посылают сообщение Commit, сравнивается значение хеша *hvi*. Тот, у кого значение *hvi* будет больше, принимает роль инициатора.

Протоколом предусмотрена повторная передача данного сообщения до 10 раз, после чего также принимается решение о невозможности продолжать выполнение протокола ZRTP и устанавливать сессию в защищенном режиме. Повторная передача сообщения Commit производится с задержкой, величина которой имеет переменное значение 150, 300, 600, 1200 мс, причем, начиная с четвертой передачи, задержка имеет постоянную величину 1200 мс. Каждое принятое сообщение Commit подтверждается сообщением DHPart1, после приема которого передача Commit прекращается.

В результате обмена открытыми сообщениями DHPart1 и DHPart2 формируются секретные ключи для SRTP-сессии. При этом для защиты от атаки MITM-протокол позволяет использовать данные предыдущих соединений. Для этого используется специальная таблица в памяти устройств, поддерживающих ZRTP-протокол, индексированная по ZID респондента.

Оба корреспондента используют полученные открытые ключи *pvi* и *pvr* для расчета результирующего ключа обмена Диффи — Хелмана. При

обмене в сообщениях передаются рассчитанный ранее *pvi/pvr* и хеш-функции значений регистров данных, распределенных в предыдущей сессии.

Корреспонденты сверяют полученные в сообщениях значения регистров данных, распределенных в предыдущей сессии, со значениями, рассчитанными локально.

Протоколом предусмотрена повторная передача сообщения DHPart2 до 10 раз, после чего также принимается решение о невозможности продолжать выполнение протокола ZRTP и устанавливать сессию в защищенном режиме. Повторная передача сообщения DHPart2 производится с задержкой, величина которой имеет переменное значение 150, 300, 600, 1200 мс, причем, начиная с четвертой передачи, задержка имеет постоянную величину 1200 мс. Каждое принятое сообщение DHPart2 подтверждается сообщением Confirm, после приема которого передача DHPart2 прекращается.

Для подтверждения успешного формирования секретных ключей происходит обмен сообщениями Confirm1 и Confirm2, которые содержат зашифрованное с помощью Advanced Encryption Standard в режиме Cipher FeedBack (AES CFB) сообщение, передающее несколько флагов и параметров, включая время действия нового сгенерированного ключа, а также некоторые служебные флаги и опциональные цифровые подписи. Для шифрования используются ключи, рассчитанные на предыдущей фазе протокола.

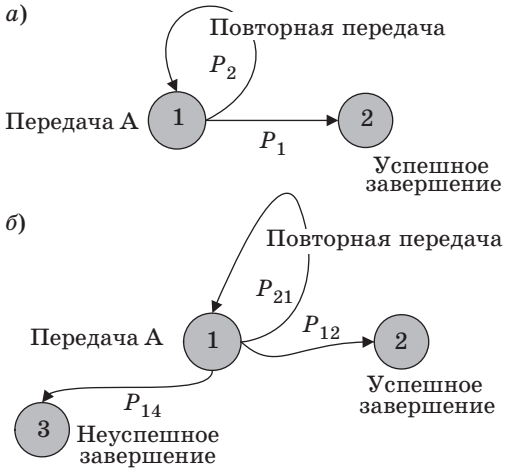
Протокол считается завершенным, когда респондент отправляет сообщение Conf2ACT или первый SRTP-пакет с верным тегом аутентификации.

Математическая модель протокола и оценка вероятностно-временных характеристик

Исследуя работу протокола ZRTP в канале связи с ошибками, имеет смысл оценивать такие характеристики протокола, как зависимость среднего времени выполнения от вероятности ошибки в канале связи и вероятность успешного завершения протокола. Анализ проводится для дискретного канала без памяти, параметром которого является вероятность ошибки p_0.

Для анализа вероятностно-временных характеристик (ВВХ) протокола целесообразно использовать апробированный метод вероятностных графов [8].

Граф, описывающий переход из начального состояния протокола ZRTP в состояние завершения первого этапа первой фазы при неограниченном числе передач одного сообщения Hello, имеет вид, представленный на рис. 4, *а*. Производящая функция такого вероятностного графа



■ ***Рис. 4.*** Граф передачи сообщения Hello при бесконечном (*а*) и конечном (*б*) числе повторов

$$f(z) = \sum_{i=1}^{v} P_i Z^{t_i},$$

где P_i — вероятность перехода из первого состояния во второе; t_i — время, необходимое для перехода.

При расчете времени перехода от одной вершины графа к другой необходимо учитывать как время передачи сообщения протокола передачи, так и задержки, вызванные ожиданием перед передачей сообщения, которые также зависят от номера итерации, при этом производящие функции переходов имеют вид:

$$f_1 = p_1 x^{a1},$$

где p_1 — вероятность успешного приема сообщения респондентом; a_1 — время передачи сообщения, которое определяется длиной передаваемого сообщения и скоростью передачи;

$$f_2 = p_2 x^{a2},$$

где p_2 — вероятность неуспешного приема сообщения респондентом; a_2 — время ожидания перед повторной передачей сообщения.

Тогда производящая функция графа

$$f_{12p} = p_1 x^{a1} + p_1 x^{a1} p_2 x^{a2} + \ldots + p_1 x^{a1} (p_2 x^{a2})^i + \ldots =$$

$$= f_1 + f_1 f_2 + \ldots + f_1 f_2^i + \ldots = \frac{f_1}{1 - f_q}.$$

Однако в протоколе с конечным числом передач в графе (см. рис. 4, *а*) добавляется третье состояние неуспешного завершения (рис. 4, *б*).

Определим производящую функцию, учитывающую конечное число повторов, равное k. Поскольку общая производящая функция имеет вид

$$f_{12p} = p_1 x^{a1} + p_1 x^{a1} p_2 x^{a2} + \ldots + p_1 x^{a1} (p_2 x^{a2})^k +$$

$$+ (p_1 x^{a2})(p_2 x^{a2})^{k+1} \ldots + p_1 x^{a1} (p_2 x^{a2})^i + \ldots,$$

то производящая функция перехода из точки *1* в точку *2* записывается как

$$f_{12} = p_1 x^{a1}\left(1 + p_2 x^{a2} + (p_2 x^{a2})^2 + \ldots + (p_2 x^{a2})^k\right).$$

В случае с одинаковым временем задержки повтора

$$f_{12} = p x^{a1} \sum_{i=0}^{k} (p_2 x^{a2})^i,$$

сумма $k + 1$ первых членов может быть также определена как

$$f_{12} = \frac{f_1(f_2^{k+1} - 1)}{(f_2 - 1)}. \quad (1)$$

Тогда производящая функция перехода из точки *1* в точку *3*

$$f_{13} = p_1 x^{a1}(p_2 x^{a2})^{k+1} + \ldots + p_1 x^{a1}(p_2 x^{a2})^i + \ldots =$$
$$= p_1 x^{a1}(p_2 x^{a2})^{k+1}\left(1 + \ldots + 1(p_2 x^{a2})^{i-(k+1)} + \ldots\right) =$$
$$= p_1 x^{a1} \frac{(p_2 x^{a2})^{k+1}}{1 - p_2 x^{a2}} = \frac{f_2^{k+1} f_1}{(1 - f_2)}.$$

Среднее время перехода из точки *1* в точку *2* определяется соотношением

$$T_{\text{ñð}} = \frac{\mathrm{d}f_{12}}{\mathrm{d}x}(x = 1). \quad (2)$$

Рассмотрим случай, когда времена ожидания при повторе сообщения a_{2i} будут разными. Производящая функция при этом будет иметь вид

$$f_{12} = p_1 x^{a1} + p_1 x^{a1} p_2 x^{a21} + \ldots + p_1 x^{a1}(p_2 x^{a2k})^k +$$
$$+ p_1 x^{a1}(p_2 x^{a2(k+1)})^{k+1} \ldots p_1 x^{a1}(p_2 x^{a2(k+i)})^{k+i}.$$

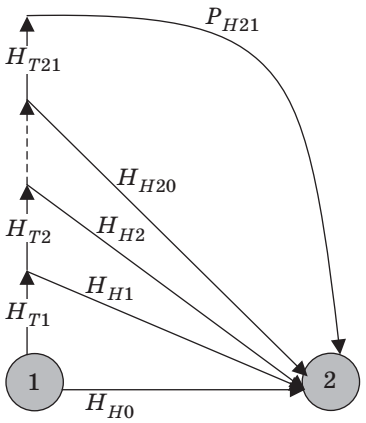
Поскольку при переменном времени эта производящая функция не является геометрическим рядом, к ней не применимо соотношение (1).

Пусть времена $a_{21} \ldots a_{2(m-1)}$ выполнения повторов 1 ... $(m - 1)$ разные, $a_{2m} \ldots a_{2k}$ — одинаковые. Тогда производящая функция перехода из точки *1* в точку *2* определяется выражением

$$f_{12} = p_1 x^{a1} \sum_{i=1}^{m-1} (p_2 x^{a2i})^i +$$
$$+ p_1 x^{a1}(p_2 x^{a2m})^m \frac{\left(\left\lfloor p_2 x^{a2m} \right\rfloor^{k-m+1} - 1\right)}{(p_2 x^{a2m} - 1)}. \quad (3)$$

Составим полный вероятностный граф для первой фазы протокола ZRTP с переменным временем ожидания (рис. 5) и определим следующие параметры:

T_{HA} — время формирования и передачи сообщения Hello корреспондентом A;



■ ***Рис. 5.*** Полный вероятностный граф передачи сообщения Hello

$T_{\text{ож}}$ — время ожидания при передаче сообщения Hello, которое выжидает корреспондент между повторными передачами сообщения;

l — размер пакета, передаваемого по каналу связи;

p_0 — вероятность битовой ошибки в канале связи.

При определении производящей функции передачи одного сообщения Hello первой фазы протокола учтем следующие особенности:

— повтор сообщения Hello производится только 20 раз, причем $T_{\text{ож}}$ будет изменяться по закону

$$T_{Hi}(i) = \begin{cases} 0\ \text{ïðè}\ i = 0 \\ 0{,}05\ \text{ïðè}\ i = 1 \\ 0{,}1\ \text{ïðè}\ i = 2 \\ 0{,}2\ \text{ïðè}\ i \geq 3 \end{cases}.$$

Также введем допущение, что доставка сообщения Hello не подтверждается сообщением HelloACK.

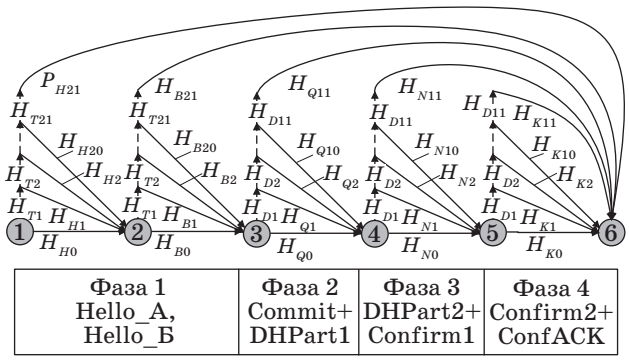
При таких условиях производящая функция первой фазы протокола передачи одного сообщения Hello будет иметь вид

$$F(p_0) = H_{H0} + H_{H1} + \ldots + H_{H20} + H_{\text{í}21}, \quad (4)$$

где $H_{H0} = \left(1 - (1 - p_0)^l\right) x^{T_{HA}}$ — производящая функция ветви безошибочной передачи сообщения Hello при первой передаче сообщения; $T_{HA} = l/c$ (c — скорость канала связи);

$H_{H1} = \left(1 - (1 - p_0)^l\right)^2 x^{T_{HA} + T_{\text{îæ}}}$ — производящая функция, определяющая ветвь безошибочной передачи сообщения Hello при первой повторной передаче;

$H_{H(i)} = \left(1 - (1 - p_0)^l\right)^i x^{T_{HA} + iT_{\text{îæ}}}$ — производящая функция, определяющая ветвь безошибочной передачи сообщения Hello при i-й повторной передаче сообщения;



■ ***Рис. 6.*** Вероятностный граф протокола ZRTP

$H_{H21} = P_{\text{îñò}} x^{T_{HA}+20T_{\text{îæ}}}$ — производящая функция, определяющая ветвь безуспешной передачи сообщения после 20 повторных передач сообщения, где $P_{\text{ост}}$ отражает вероятность, что сообщение Hello не было передано за 20 попыток.

Для составления полного вероятностного графа работы протокола ZRTP (рис. 6) воспользуемся диаграммой взаимодействия (см. рис. 3).

Для получения необходимых оценок в соответствии со статьей [8] граф разбивается на ветви, характеризующие успешное и неуспешное выполнение протокола в каждой фазе и затем упрощается для определения полной производящей функции протокола. Производящая функция упрощенного графа (рис. 7) используется для оценки среднего времени успешного завершения протокола.

Производящая функция всего графа имеет вид

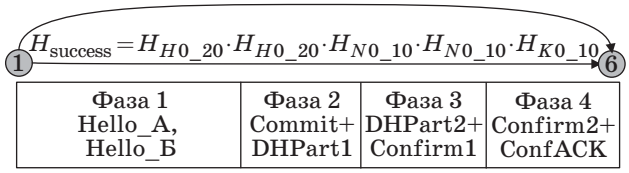
$$H_{\text{full}} = H_{H0\_20} \cdot H_{H0\_20} \cdot H_{Q0\_10} \cdot H_{N0\_10} \times \times H_{K0\_10} + H_{T21} \cdot H_{H21} + H_{H0\_20} \times \times (H_{T21} \cdot H_{H21} + H_{H0\_20} \cdot H_{D11}[H_{Q11} + + H_{Q0\_10} \cdot (H_{N0\_10} \cdot H_{K11} + H_{N11})]).$$

Производящая функция ветви успешного завершения протокола имеет вид

$$H_{\text{success}} = H_{H0\_20} \cdot H_{H0\_20} \times \times H_{Q0\_10} \cdot H_{N0\_10} \cdot H_{K0\_10}. \quad (5)$$

Производящая функция ветви неуспешного завершения протокола имеет вид

$$H_{\text{fail}} = H_{T21} \cdot H_{H21} + H_{H0\_20} \times \times (H_{T21} \cdot H_{H21} + H_{H0\_20} \cdot H_{Q11}[H_{Q11} + + H_{Q0\_10} \cdot (H_{N0\_10} \cdot H_{K11} + H_{N11})])$$



■ ***Рис. 7.*** Упрощенный вероятностный граф протокола ZRTP

$$H_{\text{fail}} = H_{T21} \cdot H_{H21} + H_{H0\_20} \cdot (H_{T21} \cdot H_{H21} + + H_{H0\_20} \cdot H_{D11}[H_{Q11} + H_{Q0\_10} \times \times (H_{N0\_10} \cdot H_{K11} + H_{N11})]).$$

На первом и втором этапе первой фазы протокола ZRTP, при передаче Hello от корреспондента А к корреспонденту Б и при передаче Hello от Б к А, сообщения имеют одинаковые параметры, поэтому передача сообщений для обоих этапов может быть представлена в виде одинаковых функций:

$$H_{H0\_20}(x, p_0) = \left[\sum_{y=0}^{19}\left[H_A(y, x, p_0)\left(\prod_{i=1}^{y} H_T(i, x)\right)\right]\right] + + H_{A\_\text{îñò}}(x, p_0)\prod_{i=0}^{20} H_T(i, x),$$

где производящая функция передачи сообщения с номером y

$$H_A(y,x,p_0) = \left[(1-p_0)^{NH} x^{\frac{NH}{c}}\left[\left[1-(1-p_0)^{NH}\right] x^{T_{Hi}(y)}\right]^y\right];$$

$$H_{A\_\text{îñò}}(x,p_0) = 1 - \left[\sum_{y=0}^{19}\left[\left[(1-p_0)\right]^{NH}\left[\left[1-(1-p_0)^{NH}\right]^y\right]\right]\right].$$

По аналогии определяются производящие функции для следующих фаз протокола: H_{Q0_10} и H_{N0_10} описывают передачу сообщений второй и третьей фазы протокола, а именно передачу Commit+DH1 и DH2+Confirm1 сообщений. Общая максимальная длина сообщений Commit+DH1 и DH2+Confirm1 составляет соответственно 5600 и 5280 бит. В связи с тем, что в фазах 2 — 4 определены иные времена ожидания, используется производящая функция H_D ветви ожидания перед повторной передачей сообщения в случае недоставки на предыдущей попытке:

$$H_D(i, x) = x^{T_{Di(i)}},$$

$$T_{Di}(i) = \begin{cases} 0 \text{ ïðè } i = 0 \\ 0{,}15 \text{ ïðè } i = 1 \\ 0{,}3 \text{ ïðè } i = 2 \\ 0{,}6 \text{ ïðè } i = 3 \\ 1{,}2 \text{ ïðè } i \geq 4 \end{cases}.$$

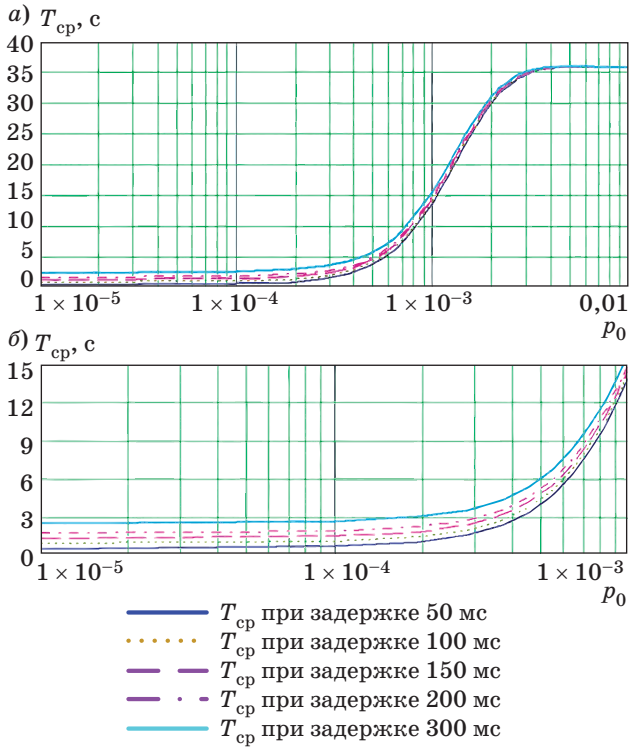
Аналогично определяется производящая функция H_K для четвертой фазы протокола, когда передаются сообщения Confirm2 + Confirm2ACK общим размером 1632 бит.

Для расчета среднего времени успешного завершения протокола в соответствии с (2) необходимо вычислить первую производную производя-

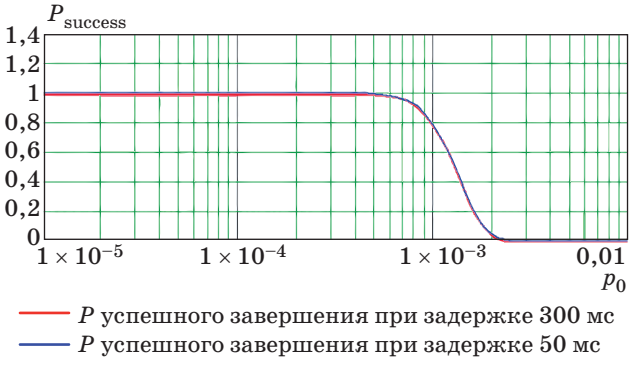
щей функции ветви успешного выполнения протокола (5) в точке $x = 1$.

Для вышеописанной модели был произведен расчет для работы протокола в каналах связи с разным значением вероятности ошибок и при различных величинах задержки пакетов. График зависимости среднего успешного времени выполнения протокола ZRTP представлен на рис. 8, *а*, а в канале с малой вероятностью ошибки — на рис. 8, *б*.

Вероятность успешного завершения определяется соотношением



■ ***Рис. 8.*** Зависимость среднего времени выполнения протокола ZRTP от вероятности ошибки в канале с задержками пакетов на интервале $10^{-5} \le p_0 \le 10^{-2}$ (*а*) и $10^{-5} \le p_0 \le 10^{-3}$ (*б*)



■ ***Рис. 9.*** Зависимость вероятности успешного завершения протокола ZRTP от вероятности ошибки

$$P_{success} = H_{H0\_20}(x=1)H_{H0\_20}(x=1)\times$$
$$\times H_{Q0\_10}(x=1)H_{N0\_10}(x=1)H_{K0\_10}(x=1).$$

График зависимости вероятности успешного завершения протокола от вероятности ошибок в канале представлен на рис. 9.

Экспериментальная оценка ВВХ протокола

Целью эксперимента является оценка влияния потери пакетов в канале передачи данных и задержки данных при передаче на качество речи, а также оценка среднего времени работы и вероятности успешного завершения протокола ZRTP.

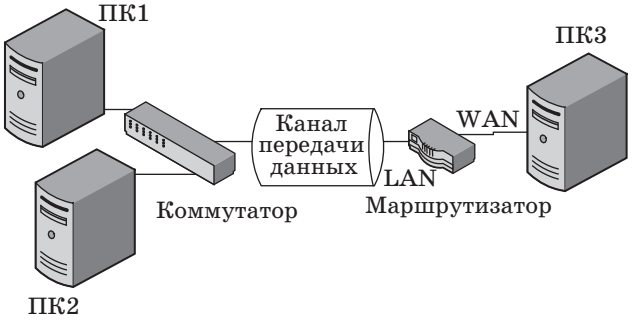
Экспериментальная установка (рис. 10) состоит из трех компьютеров, управляемого коммутатора, маршрутизатора на базе ОС FreeBSD. Маршрутизатор выполняет роль имитатора канала передачи данных, воспроизводя различные состояния канала связи, и позволяет задавать величины двух параметров канала связи: процента потерянных пакетов, передаваемых через порт маршрутизатора, а также задержки для передаваемых пакетов.

Для тестирования использовались персональные компьютеры.

На компьютере ПК1 установлено ПО Wireshark, позволяющее перехватывать и анализировать пакеты, передаваемые между ПК2 и ПК3. Для реализации этой цели на управляемом коммутаторе включается функция зеркалирования портов.

На ПК2 и ПК3 устанавливаются программы — VoIP-клиент Phoner и VoIP-шлюз Zfone. Программа Phoner была выбрана в качестве VoIP-клиента, так как она имеет встроенную, настраиваемую поддержку ZRTP-протокола. Эта опция позволяет включать/выключать встроенный модуль ZRTP при проведении тестов.

Zfone — программа, созданная Филипом Циммерманом, разработчиком протокола ZRTP. Фактически данная программа играет роль шлюза для RTP-пакетов, преобразуя их в SRTP, а также позволяет выполнять ZRTP-протокол между корреспондентами для генерации ключей. В случае



■ ***Рис. 10.*** Схема экспериментальной установки

выключения опции поддержки ZRTP на Phoner и запуска программы Zfone ZRTP-протокол будет выполняться средствами программы Zfone, что позволит сравнить поведение протокола ZRTP в реализациях двух разных разработчиков.

Перед измерением на интерфейсе маршрутизатора устанавливались настройки: процент потерянных пакетов, задержка для передаваемых пакетов.

На ПК3 в программе Phoner был включен режим автоподнятия трубки с воспроизведением тестовой записи. При звонке с ПК2 на ПК3 на ПК2 автоматически включалась запись разговора и сохранялась тестовая запись в том виде и с тем качеством, с которым она была доступна пользователю.

Измерение проводилось в следующей последовательности:

1) устанавливались требуемые величины потери пакетов и задержки канала передачи данных на маршрутизаторе;

2) выполнялась проверка точности установки задержки и потери пакетов утилитой ping;

3) включался сетевой снифер на ПК1, совершался звонок ПК2–ПК3, сохранялся дамп и запись звонка;

4) по дампу определялось время работы ZRTP.

Как отмечалось ранее, работа протокола ZRTP может быть организована одним из двух способов:

1) параллельно с RTP, т. е. до окончания работы ZRTP RTP-трафик передается в открытом виде. По окончании работы ZRTP голосовой трафик передается зашифрованным в SRTP;

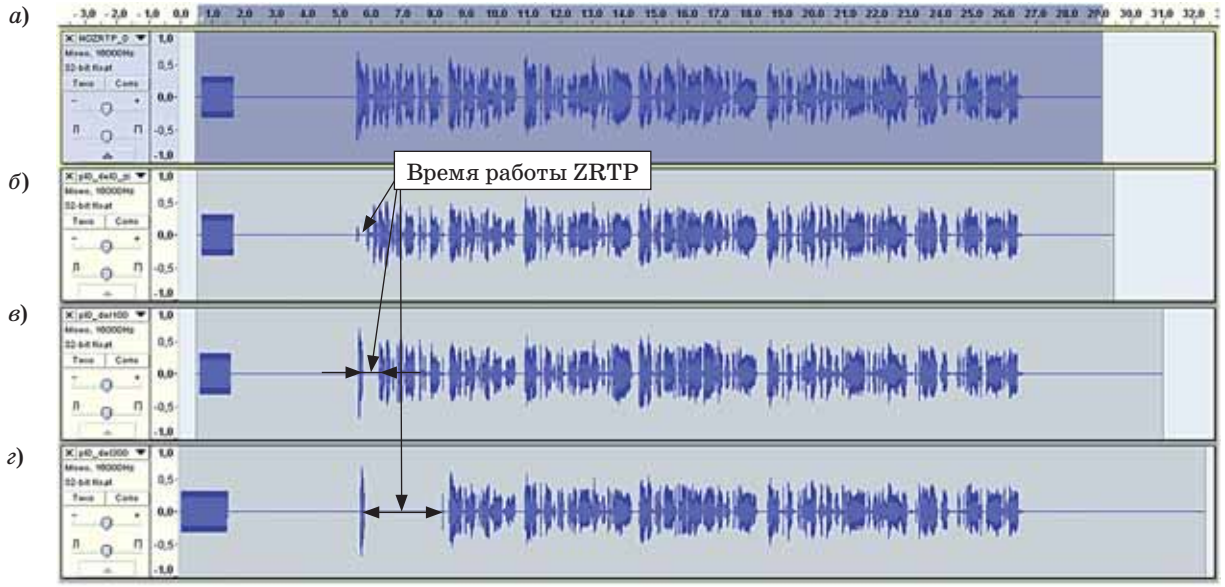
2) до RTP; как только между абонентами включается голосовой канал, ZRTP начинает работать, при этом блокируется передача RTP. Разговор между абонентами начинается по окончании ZRTP с использованием SRTP.

Программа Zfone реализует второй способ, поэтому работу ZRTP можно оценить также по голосовым диаграммам.

В качестве источника эталонного речевого сигнала (рис. 11) на одном из компьютеров был включен автоответчик, который диктует фразу «*Нажмите 1, чтобы принять этот звонок; нажмите 2, чтобы отклонить его; нажмите 3, чтобы всегда принимать звонки с этого номера; нажмите 4, чтобы никогда не принимать звонки с этого номера; нажмите 5, чтобы сбрасывать звонки с этого номера и сообщить звонящему, чтобы он внес вас в список абонентов, „кому не надо звонить“*».

На втором компьютере производилась запись принятого речевого сигнала.

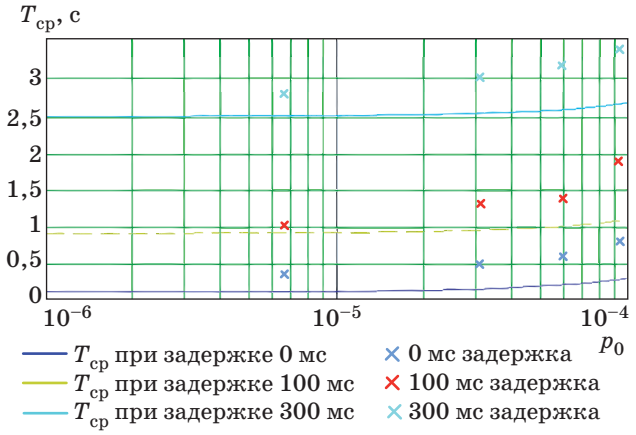
■ ***Рис. 11.*** Запись исходного речевого сообщения



■ ***Рис. 12.*** Записи принятого речевого сообщения с учетом влияния протокола ZRTP в реализации Zfone: *а* — при выключенном протоколе ZRTP; *б* — при передаче по каналу с односторонней задержкой 0 мс; *в* — при передаче по каналу с односторонней задержкой 100 мс; *г* — при передаче по каналу с односторонней задержкой 300 мс

■ ***Таблица 2.*** Результаты измерения времени выполнения протокола ZRTP

Задержка, мс	Время выполнения, с, при потерях, %			
	1	5	10	15
0	0,26	0,42	0,5	0,71
100	0,99	1,19	1,38	1,90
300	2,89	3,08	3,19	3,4
p_0	$5{,}8 \cdot 10^{-6}$	$2{,}9 \cdot 10^{-5}$	$6 \cdot 10^{-5}$	$9{,}3 \cdot 10^{-5}$



■ ***Рис. 13.*** Сравнение теоретических расчетов и результатов измерений среднего времени работы протокола ZRTP

Голосовые дорожки при передаче тестового звука по схеме компьютер-компьютер с учетом работы протокола ZRTP показаны на рис. 12, *а*—*г*.

Результаты измерений для различных задержек и потери пакетов в канале связи приведены в табл. 2. Для устранения случайных погрешностей в таблицу вносились средние значения измеряемого времени по результатам 20 измерений. Очевидно, что в каналах связи с задержками время выполнения протокола может возрастать до нескольких секунд. Например, для канала связи с задержкой 300 мс время выполнения протокола составляет около 3 с. При этом время начала защищенной передачи голоса между абонентами сдвигается на величину, равную времени работы протокола.

На рис. 13 представлены результаты сравнения времени выполнения протокола, полученные на основании теоретического анализа и экспериментальной оценки.

Заключение

Исследование показывает, что среднее время работы протокола ZRTP определяется в основном величиной задержки сообщений в канале связи. Зависимость среднего времени работы протокола ZRTP от вероятности битовых ошибок в каналах хорошего и удовлетворительного качества слабо выражена, но существенно возрастает при увеличении вероятности ошибок свыше $1 \cdot 10^{-4}$.

Результаты экспериментальной оценки среднего времени работы протокола ZRTP подтверждают теоретические расчеты. Вместе с тем они несколько превышают расчетные величины, что обусловлено упрощениями, введенными при построении математической модели. Среднее время работы протокола ZRTP в каналах с задержкой более 200 мс превышает 1,5 с и в зависимости от используемого режима защищенной IP-телефонии приводит либо к потере соответствующего фрагмента речевого сообщения, либо к передаче этого фрагмента речи в незащищенном режиме.

Таким образом, имеются существенные предпосылки для совершенствования протокола ZRTP с целью снизить зависимости времени работы протокола от задержки в канале.

Литература

1. **RFC 3261** Session Initiation Protocol, 2002. http://tools.ietf.org/html/rfc3261 (дата обращения: 18.09.2012).
2. **RFC 3550**. A Transport Protocol for Real-Time Applications, 2003. http://www.ietf.org/rfc/rfc3550.txt (дата обращения: 22.09.2012).
3. **RFC 3711**. The Secure Real-time Transport Protocol (SRTP), 2004. http://www.ietf.org/rfc/rfc3711.txt (дата обращения: 10.09.2012).
4. **Menezes P. van Oorschot, S. Vanstone.** Handbook of Applied Cryptography // CRC Press. 1996. P. 515–520, 522–524.
5. **Bresciani R., Butterfield A.** Formal security proof for the ZRTP Protocol // Intern. Conf. for Internet Technology and Secured Transactions, London, 9–12 Nov. 2009. ICITST, 2009. P. 1–6.
6. **RFC 6189**. ZRTP: Media Path Key Agreement for Unicast Secure, 2011. http://tools.ietf.org/html/rfc6189 (дата обращения: 20.09.2012).
7. **Ковцур М. М., Никитин В. Н., Юркин Д. В.** Протоколы обеспечения безопасности VoIP-телефонии // Защита информации. Инсайд. 2012. № 3. С. 74–81.
8. **Никитин В. Н., Юркин Д. В.** Улучшение способов аутентификации для каналов связи с ошибками // Информационно-управляющие системы. 2010. № 6. С. 42–46.