МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 004.056 DOI 10.17223/20710410/66/6

АТАКИ НА ПРОТОКОЛЫ АУТЕНТИФИЦИРОВАННОЙ ВЫРАБОТКИ ОБЩЕГО КЛЮЧА ПРИ НАВЯЗЫВАНИИ БУДУЩИХ ОТКРЫТЫХ ЭФЕМЕРНЫХ КЛЮЧЕЙ

Е. К. Алексеев, С. Н. Кяжин, С. В. Смышляев

ООО «КРИПТО-ПРО», г. Москва, Россия

E-mail: alekseev@cryptopro.ru, kyazhin@cryptopro.ru, svs@cryptopro.ru

Исследуется построение атак на протоколы аутентифицированной выработки общего ключа при наличии у нарушителя возможности навязывать участнику использование эфемерных открытых значений. Обосновывается актуальность рассмотрения указанной возможности. Описываются атаки на протоколы SIGMA, SIGMA-R, STS-MAC, «Эхинацея-3» и постквантовый протокол BKM-KK. Приводятся рассуждения о конструктивных особенностях протоколов, позволяющих защититься от атак такого типа.

Ключевые слова: *криптография, криптографический протокол, аутентифицированная выработка общего ключа, атака, навязывание открытых эфемерных ключей.*

FORCING FUTURE PUBLIC EPHEMERAL KEYS TO ATTACK AUTHENTICATED KEY ESTABLISHMENT PROTOCOLS

E. K. Alekseev, S. N. Kyazhin, S. V. Smyshlyaev

CryptoPro LLC, Moscow, Russia

This paper studies the security of the authenticated key establishment protocols against the adversary who has the capability to force the participants to use of ephemeral public values. The paper substantiates the relevance of considering this capability, describes, in particular, attacks on the SIGMA, SIGMA-R, STS-MAC, Echinacea-3 protocols and the post-quantum BKM-KK protocol, and discusses the design features of protocols that allow to protect against attacks of this type.

Keywords: *cryptography, cryptographic protocol, authenticated key establishment, attack, forcing public ephemeral keys.*

Введение

Одним из первых шагов при проведении криптографического анализа или синтеза любой системы является определение потенциальных возможностей нарушителя по взаимодействию с системой на качественном уровне. Важности этого этапа посвящён ряд как иностранных, так и отечественных работ [1–4].

В работе [5] рассматриваются вопросы определения возможностей нарушителя для протоколов аутентифицированной выработки общего ключа (Authenticated Key Establishment, AKE) и впервые, насколько известно авторам, упоминается возможность нарушителя навязывать открытые эфемерные значения, которые стороны будут использовать при будущем взаимодействии. Подчеркнём, что данная возможность подразумевает подмену значения открытого эфемерного ключа не в процессе передачи по каналу связи, а до начала взаимодействия, непосредственно на стороне участника, который использует его при выполнении протокола, считая корректным элементом эфемерной ключевой пары. При этом в указанных работах данная возможность нарушителя не исследуется ни в части её актуальности на практике, ни в части уязвимости к ней известных AKE-протоколов, ни в части подходов к обеспечению защиты от атак, основанных на её применении.

В настоящей работе приведены следующие результаты первых шагов по исследованию такой возможности нарушителя. В п. 1 анализируется актуальность такой возможности на практике. Показывается, что построение протоколов, стойких по отношению к нарушителям с такими возможностями, позволит улучшить эффективность протоколов и снизить требования к защищённому хранению данных сторонами взаимодействия. Указанные улучшения могут стать особенно актуальными для случая применения постквантовых криптографических алгоритмов, открытые ключи которых зачастую в десятки раз больше ключей классических механизмов. В п. 2 приводятся атаки на такие известные протоколы, как SIGMA, SIGMA-R, STS-MAC, «Эхинацея-3» и постквантовый протокол BKM-KK. В п. 3 обсуждаются вопросы построения AKE-протоколов, защищённых от атак со стороны нарушителей, которые обладают возможностью навязывать сторонам открытые эфемерные значения.

1. Актуальность рассмотрения возможности навязывания открытых эфемерных значений

Источником возможностей нарушителя, в первую очередь, является порядок применения криптографического механизма в более высокоуровневой системе или условия его эксплуатации на практике. Если не учесть какую-либо существенную возможность, можно получить расхождение прогноза о стойкости, сформированного в результате проведения криптографического анализа, с действительностью, как, например, получилось когда-то с подпротоколом Record протокола TLS 1.0 [6, 7]. Процесс выявления потенциальных возможностей нарушителя, по всей видимости, невозможно формализовать, и его успешность зависит лишь от опыта работающих аналитиков.

При этом в процессе синтеза криптосистемы целесообразно наделять потенциального нарушителя максимально широким спектром возможностей. Действительно, чем больше возможностей у нарушителя, относительно которого удалось обосновать стойкость новой криптосистемы, тем меньше требований будет предъявлено к порядку её эксплуатации. Например, если криптосистема стойка относительно нарушителя, который может узнавать какие-то промежуточные значения, возникающие при реализации протокола, то участники могут не беспокоиться о безопасном удалении этих значений. Таким образом, некоторые возможности, которые в настоящее время кажутся нереальными на практике, при их рассмотрении могут открывать существенные перспективы в части создания эффективных криптосистем. Возможность нарушителя, которой посвящена настоящая работа, относится именно к таким.

Любой современный AKE-протокол предполагает использование участниками эфемерных, то есть одноразовых, значений. Это могут быть и случайно сгенерированные

битовые строки (например, значения client_random и server_random в протоколах TLS Handshake разных версий), и более сложные объекты, как, например, эфемерные ключевые пары, используемые для выработки общего секрета по схеме Диффи — Хеллмана. В большинстве реализаций такие параметры порождаются с помощью различного типа датчиков случайных чисел непосредственно в процессе взаимодействия по протоколу. Однако стойкие современные AKE-протоколы зачастую требуют выполнения большого объёма вычислений, что при их реализации на низкоресурсных устройствах приводит к поиску путей для оптимизации. Одним из таких путей, который явно нарушает изначальную конструкцию протокола, является использование эфемерных ключевых пар в нескольких сеансах. Про такой сценарий написан отдельный раздел 2.12 документа [8], определяющего протокол IKEv2, он обсуждается также в [9]. Такая оптимизация предполагает, что эфемерный ключ должен где-то храниться длительное (по сравнению с выполнением одного сеанса протокола) время. При этом храниться ключевая пара должна в защищённой памяти, размер которой также может быть ограничен на низкоресурсных устройствах. Таким образом, следующий шаг оптимизации состоит в том, чтобы хранить защищённо лишь закрытую часть эфемерного ключа. При этом у нарушителя может быть возможность не только узнать соответствующую открытую его часть (такая возможность уже рассматривалась в работе [10]), но и подменить её. Отметим, что работа носит теоретический характер, в ней не рассматриваются практические сценарии навязывания (подмены) сеансовой ключевой информации. Другим подходом к оптимизации вычислений, который нарушает спецификацию протокола в меньшей степени, является предварительное вычисление какого-то количества эфемерных ключей для будущих сеансов. В таком сценарии у низкоресурсных устройств тем более может возникнуть соблазн хранить хотя бы открытые части в памяти, доступной для нарушителя.

С другой стороны, если какой-нибудь протокол является стойким (в требуемом от него смысле) относительно нарушителя, который может навязывать сторонам открытые эфемерные значения, то при его реализации использование предвычисленных значений, хранящихся в памяти, доступной для нарушителя, является вполне безопасным путём оптимизации. С переходом на постквантовые криптографические алгоритмы такая возможность может стать востребованной не только низкоресурсными устройствами, но и более привычными платформами. Это объясняется тем, что размеры открытых ключей некоторых алгоритмов достигают сотен тысяч байтов при том, что открытые ключи классических алгоритмов, основанных на эллиптических кривых, занимают несколько десятков байтов. Ниже мы отдельно рассуждаем о стойкости протоколов, основанных на постквантовых алгоритмах, и приводим пример атаки на один из таких протоколов.

2. Атаки на известные AKE-протоколы

Опишем атаки на известные AKE-протоколы, в ходе которых нарушитель навязывает участникам взаимодействия открытые эфемерные значения. Рассматриваются протоколы из трёх классов, которые определяются типом ключа, используемым для аутентификации сторон:

1) ключ подписи;
2) скаляр (используемый при вычислениях, задаваемых непосредственно протоколом);
3) ключ механизма инкапсуляции ключа (Key Encapsulation Mechanism, KEM).

В табл. 1 перечислены описанные атаки: в первом столбце указан класс атакуемого протокола, во втором — протокол, в третьем (объединённом) — используемые возможности нарушителя («*» означает навязывание участникам взаимодействия открытых эфемерных значений без знания соответствующих им закрытых значений), в четвёртом — реализуемая угроза:

— AUTH — ложная аутентификация от имени одного участника;
— MITM — ложная аутентификация от имени двух участников;
— KCI — ложная аутентификация после вскрытия долговременного ключа проверяющего;
— SEC — нарушение секретности ключей;
— PFS — нарушение «свойства PFS», т. е. секретности ключей, выработанных до вскрытия долговременного ключа.

В последних двух столбцах приведены ссылки на описание атак.

Таблица 1

Атаки на AKE-протоколы, описанные в настоящей работе

Тип долговр. ключей	**Протокол**	**Возможности нарушителя**					**Угроза**	**Опис. атаки, разд.**	**Схема атаки, рис.**
		Навяз. откр. эфемерного ключа *инициатору*	Навяз. откр. эфемерного ключа *ответчику*	Изменение сообщений в канале	Компрометация долговременного ключа	Компрометация сеансового ключа			
Подпись	SIG-DH+	✓					AUTH	2.1	2
			✓						
	SIGMA		✓	✓					4
		✓	✓	✓			MITM		5
	STS-MAC		✓	✓			AUTH		
		✓	✓	✓			MITM		
	«Эхинацея-3»		✓	✓			AUTH		
		✓	✓	✓			MITM		
	SIGMA-R	✓		✓			AUTH		7
Скаляр	TS3	✓					AUTH	2.2	8
			✓						
	CF	✓*							10
			✓*						
	SK6	✓*			✓		KCI		12
		✓*	✓*		✓		PFS		
		✓*	✓*			✓	SEC		
KEM	BKM-KK		✓				AUTH	2.3	14

Порядок наименования и описания (в том числе обозначения, использование классов базовых криптографических механизмов без уточнения конкретных их представителей и т. д.) протоколов в целом соответствует работе [11], но для краткости протоколы описаны сразу в варианте с предвычислением эфемерных значений. Операции чтения из защищённой памяти и памяти, доступной для нарушителя, обозначены как $e \leftarrow sMEM$ и $E \leftarrow MEM$ соответственно. Важно отметить: мы считаем, что сторона во время работы по протоколу не проверяет соответствие закрытого и открытого

эфемерных ключей, так как это свело бы на нет почти всё преимущество от предварительных вычислений. Навязывание нарушителем открытых эфемерных значений указано в рамках, как действие, выполняемое соответствующей стороной. В рамках также указаны значения, изменяемые нарушителем в пересылках.

2.1. Протоколы, использующие схему подписи

Опишем атаки с навязыванием открытых эфемерных значений на протоколы SIG-DH+ [12], SIGMA [13] и SIGMA-R [13]. В результате всех указанных атак реализуется угроза ложной аутентификации.

Протокол SIG-DH+ (рис. 1).

$A : sk_A^s, pk_A^s$		$B : sk_B^s, pk_B^s$
$w_A \leftarrow sMEM$		
$W_A \leftarrow MEM$	$\xrightarrow{A, W_A}$	$w_B \leftarrow sMEM$
		$W_B \leftarrow MEM$
Verify σ_B	$\xleftarrow{B, W_B, \sigma_B}$	$\sigma_B \leftarrow \mathsf{Sig}_{sk_B^s}(B, W_B, W_A, A)$
$\sigma_A \leftarrow \mathsf{Sig}_{sk_A^s}(A, W_A, W_B, B)$	$\xrightarrow{\sigma_A}$	Verify σ_A
$Q \leftarrow w_A \cdot W_B$		$Q \leftarrow w_B \cdot W_A$
$K \leftarrow \mathsf{KDF}(Q, W_A, W_B, A, B)$		$K \leftarrow \mathsf{KDF}(Q, W_A, W_B, A, B)$
return K		**return** K

Рис. 1. Схема протокола SIG-DH+ с предвычислениями эфемерных ключей

Атака на протокол SIG-DH+ требует от нарушителя лишь навязать стороне A значение W'_A с известным нарушителю дискретным логарифмом w'_A ($W'_A = w'_A \cdot P$) вместо эфемерного ключа W_A. Передаваемые по каналу связи значения менять не требуется. При этом нарушитель может вычислить ключ K, равный тому, который вычислит сторона B, так как $Q = w_B \cdot W'_A = w'_A \cdot W_B$, а w'_A нарушителю известно. Таким образом, нарушитель вырабатывает общий ключ со стороной B, но B думает, что выработал его со стороной A. Схема атаки представлена на рис. 2.

Существенным отличием в описанном сценарии атаки от обычной замены W_A на W'_A при передаче по каналу связи является то, что при подмене в канале связи сторона A будет вычислять значение подписи σ_A от истинного значения W_A, поэтому на стороне B проверка этой подписи завершится ошибкой.

Легко видеть, что аналогичная атака работает при навязывании открытых эфемерных значений стороне B.

Уязвимыми к аналогичному сценарию атаки оказываются и другие протоколы, в которых по каналу не пересылаются значения, полученные с помощью выработанного по схеме Диффи — Хеллмана секрета (для протокола SIG-DH+ это $Q = w_A \cdot W_b = = w_b \cdot W_A$). В качестве примера можно привести протокол TS3-1 [14, 15] (упрощённое описание можно найти в [11]).

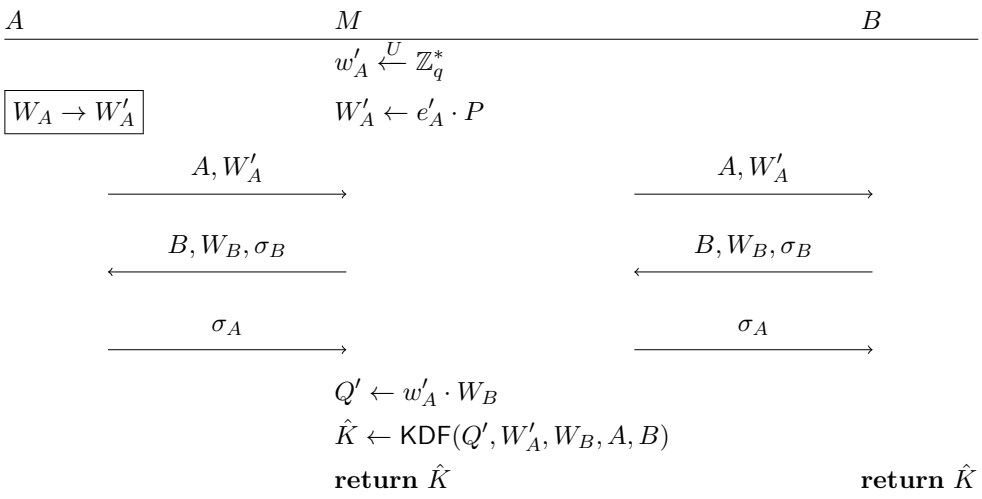

Рис. 2. Схема атаки на протокол SIG-DH+ с навязыванием эфемерных значений стороне A

Протоколы SIGMA. Перейдём к описанию атак, требующих от нарушителя не только навязывания открытых эфемерных значений, но и изменения сообщений, передаваемых по каналу связи. Напомним, что протокол SIGMA лег в основу протоколов TLS 1.3 [16], IKEv2 [8] и стандартизированного в России протокола «Эхинацея-3». Описание протокола SIGMA представлено на рис. 3.

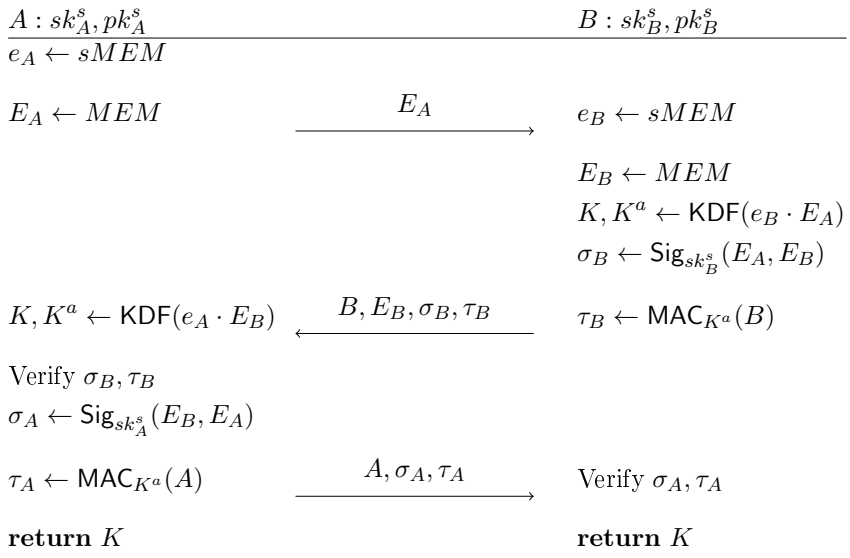

Рис. 3. Схема протокола SIGMA с предвычислениями

Пересылки, осуществляемые при реализации атаки на протокол SIGMA в варианте с навязыванием эфемерных открытых ключей стороне B, представлены на рис. 4. В результате нарушитель вырабатывает общий ключ со стороной A, а сторона A думает, что выработала его со стороной B.

Аналогичная атака при навязывании открытых эфемерных ключей стороне A не сработает, так как сторона B сформирует значение τ_B на ключе $\mathsf{KDF}(e_B \cdot E'_A)$, а сторона A будет проверять его на ключе $\mathsf{KDF}(e_A \cdot E_B)$. При этом ни значение e_A, ни

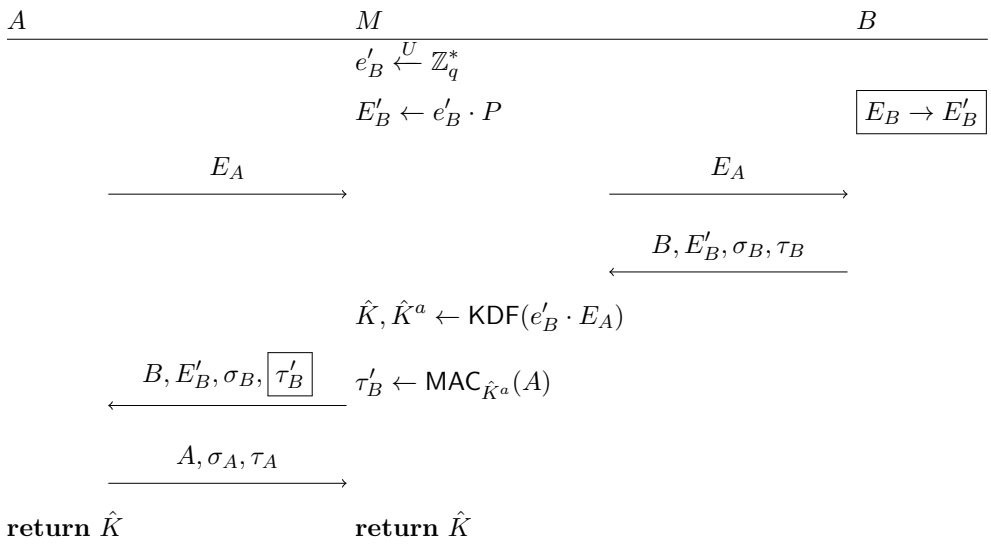

Рис. 4. Схема атаки на протокол SIGMA с навязыванием эфемерных значений стороне B

значение e_B нарушителю не известно, поэтому подменить в канале значение τ_B на то, которое успешно проверится на стороне A, он не может.

Рассмотрим атаку на протокол SIGMA при наличии у нарушителя возможности навязывать открытые эфемерные значения обеим сторонам. В этом случае получается реализовать угрозу ложной аутентификации ещё и от лица A. Схематичное описание атаки представлено на рис. 5.

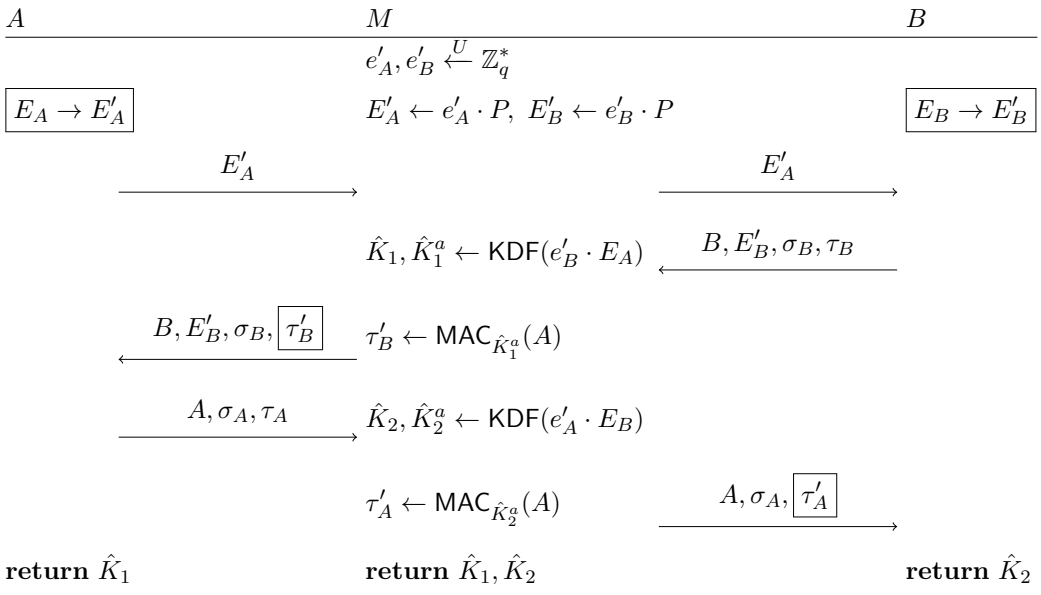

Рис. 5. Схема атаки на протокол SIGMA с навязыванием эфемерных значений двум сторонам

Протоколы STS-MAC [17] и «Эхинацея-3» [18] незначительно отличаются от протокола SIGMA (упрощённое описание можно найти в [11]), вследствие чего обе атаки на протокол SIGMA работают и для них.

Протокол SIGMA-R. Наиболее сложной с точки зрения требуемых от нарушителя действий является атака на протокол SIGMA-R, описание которого представлено на рис. 6. Сценарий атаки на протокол SIGMA-R приведён на рис. 7.

$A: sk_A^s, pk_A^s$		$B: sk_B^s, pk_B^s$
$e_A \leftarrow sMEM$		
$n_A, E_A \leftarrow MEM$	$\xrightarrow{E_A, n_A}$	$e_B \leftarrow sMEM$
$K, K_A^a, K_B^a, K_A^{ae}, K_B^{ae} \leftarrow \mathsf{KDF}(e_A \cdot E_B)$	$\xleftarrow{E_B, n_B}$	$n_B, E_B \leftarrow MEM$
$\tau_A \leftarrow \mathsf{MAC}_{K_A^a}(A)$ $\sigma_A \leftarrow \mathsf{Sig}_{sk_A^s}(n_B, E_A)$		
$C_A \leftarrow \mathcal{AE}.\mathsf{Enc}_{K_A^{ae}}(\varepsilon, (A, \sigma_A, \tau_A))$	$\xrightarrow{C_A}$	$K, K_A^a, K_B^a, K_A^{ae}, K_B^{ae} \leftarrow \mathsf{KDF}(e_B \cdot E_A)$
		$A, \sigma_A, \tau_A \leftarrow \mathcal{AE}.\mathsf{Dec}_{K_A^{ae}}(\varepsilon, C_A)$ Verify σ_A, τ_A $\tau_B \leftarrow \mathsf{MAC}_{K_B^a}(B)$ $\sigma_B \leftarrow \mathsf{Sig}_{sk_B^s}(n_A, E_B)$
$B, \sigma_B, \tau_B \leftarrow \mathcal{AE}.\mathsf{Dec}_{K_B^{ae}}(\varepsilon, C_B)$	$\xleftarrow{C_B}$	$C_B \leftarrow \mathcal{AE}.\mathsf{Enc}_{K_B^{ae}}(\varepsilon, (B, \sigma_B, \tau_B))$
Verify σ_B, τ_B **return** K		**return** K

Рис. 6. Схема протокола SIGMA-R с предвычислениями

Сценарий атаки на протокол SIGMA-R усложняется по сравнению с атакой на протокол SIGMA из-за того, что в протоколе SIGMA-R третья и четвёртая пересылки зашифровываются на ключах, выработанных из ключа Диффи — Хеллмана. При навязывании эфемерных ключей стороне A нарушителю, чтобы выдать себя за A перед B, нужно корректно сформировать вторую пересылку C_A, которую направляет сторона A и в которой содержатся значения подписи и имитовставки. Имитовставку нарушитель может посчитать сам, но подпись стороны A он должен получить из оригинального сообщения C_A, сформированного стороной A. Чтобы знать ключ, на котором будет формироваться C_A, нарушитель меняет в канале значение E_B. За счёт этого он может получить «честное» значение σ_A, а дальше уже сформировать C'_A на том ключе, на котором его будет расшифровывать сторона B.

В схематичном описании атаки при использовании функции KDF указаны только те из вырабатываемых ключей, которые нарушитель использует в дальнейшем для осуществления атаки.

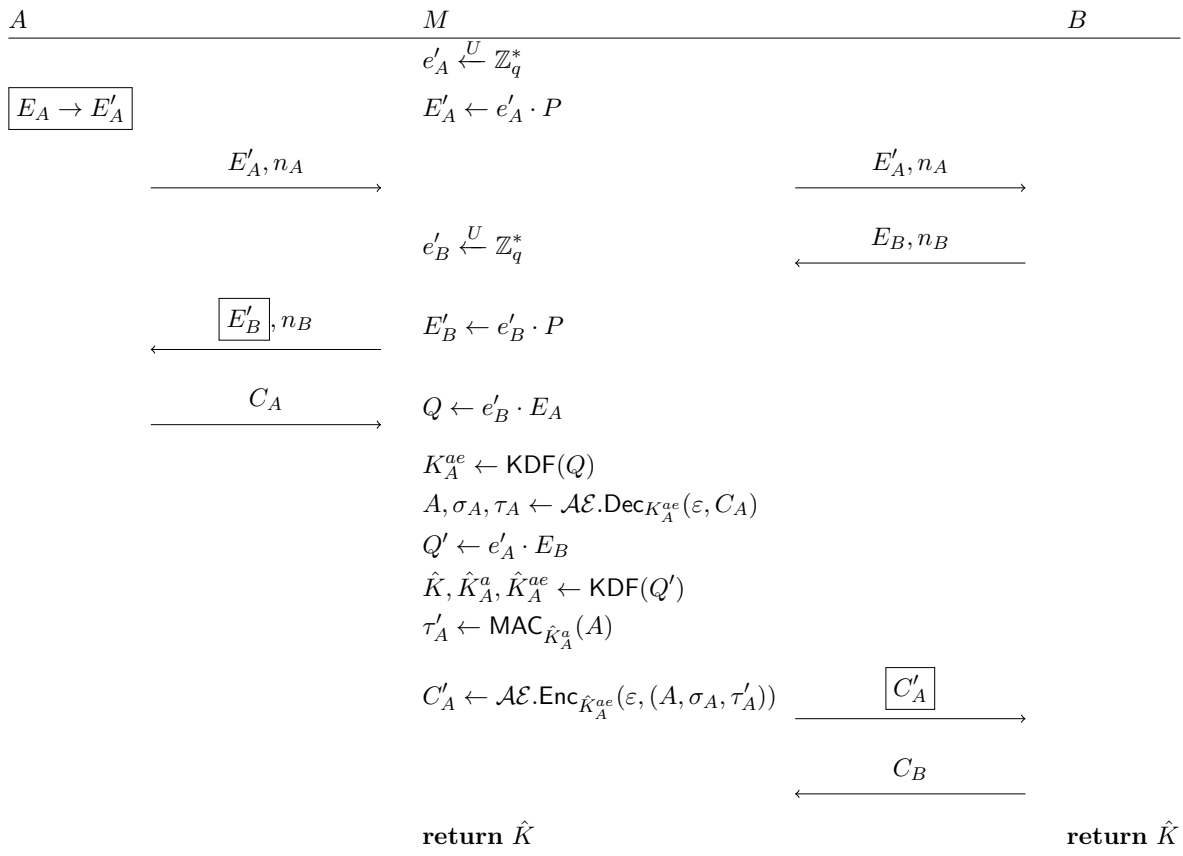

Рис. 7. Схема атаки на протокол SIGMA-R с навязыванием эфемерных значений стороне B

2.2. Протоколы, использующие долговременные скалярные величины

Опишем атаки с навязыванием открытых эфемерных значений на протоколы TS3 [14, 15], CF [19] и SK6 [20]. Для протоколов TS3 и CF атаки приводят к реализации угрозы ложной аутентификации, а для протокола SK6 — к реализации угрозы KCI и нарушению свойства PFS.

Протокол TS3 (рис. 8). Сценарий атаки такой же, как для протокола SIG-DH+, его применение становится возможным из-за того, что по каналу не пересылаются сообщения, зависящие от выработанного ключа, а аутентифицирующее сторону значение зависит лишь от идентификаторов сторон и их эфемерных ключей.

Протокол CF. Схожий сценарий работает для протокола CF, его описание представлено на рис. 9, схема атаки — на рис. 10.

Особенность сценария атаки против этого протокола в том, что нарушитель навязывает стороне A эфемерный ключ $E'_A = -X_A + P$, дискретный логарифм которого ему неизвестен. Однако за счёт того, что единственным не передаваемым по каналу связи аргументом функции выработки итогового сеансового ключа является точка $W = (e_B + x_B)(E_A + X_A)$, нарушитель может её вычислить при таком значении E'_A.

$A : x_A, X_A$		$B : x_B, X_B$
$e_A \leftarrow sMEM$		
$E_A \leftarrow MEM$		
$K^a \leftarrow \mathsf{KDF}(x_A \cdot X_B)$		
$\tau_A \leftarrow \mathsf{MAC}_{K^a}(A, B, E_A)$	$\xrightarrow{A, E_A, \tau_A}$	$K^a \leftarrow \mathsf{KDF}(x_B \cdot X_A)$
		Verify τ_A
		$e_B \leftarrow sMEM$
		$E_B \leftarrow MEM$
Verify τ_B	$\xleftarrow{B, E_B, \tau_B}$	$\tau_B \leftarrow \mathsf{MAC}_{K^a}(B, A, E_B)$
$K \leftarrow e_A \cdot E_B$		$K \leftarrow e_B \cdot E_A$
return K		**return** K

A	M	B
	$e_A \xleftarrow{U} \mathbb{Z}_q^*$	
$\boxed{E_A \to E'_A}$	$E'_A = e'_A \cdot P$	
$\xrightarrow{A, E'_A, \tau_A}$	$\xrightarrow{A, E'_A, \tau_A}$	
	$\hat{K} \leftarrow e'_A \cdot E_B$ $\xleftarrow{B, E_B, \tau_B}$	
	return $\hat{K}$	**return** $\hat{K}$

Рис. 8. Схема протокола TS3 с предвычислениями и схема атаки на него

$A : (x_A, X_A), (sk_A^s, pk_A^s)$		$B : (x_B, X_B), (sk_B^s, pk_B^s)$
$e_A \leftarrow sMEM$		
$E_A \leftarrow MEM$		
$\sigma_A \leftarrow \mathsf{Sig}_{sk_A^s}(E_A)$	$\xrightarrow{A, E_A, \sigma_A}$	Verify σ_A
		$e_B \leftarrow sMEM$
		$E_B \leftarrow MEM$
Verify σ_B	$\xleftarrow{B, E_B, \sigma_B}$	$\sigma_B \leftarrow \mathsf{Sig}_{sk_B^s}(E_B)$
$W \leftarrow (e_A + x_A) \cdot (E_B + X_B)$		$W \leftarrow (e_B + x_B) \cdot (E_A + X_A)$
$K \leftarrow \mathsf{KDF}(A, B, W, E_A)$		$K \leftarrow \mathsf{KDF}(A, B, W, E_A)$
return K		**return** K

Рис. 9. Схема протокола CF с предвычислениями

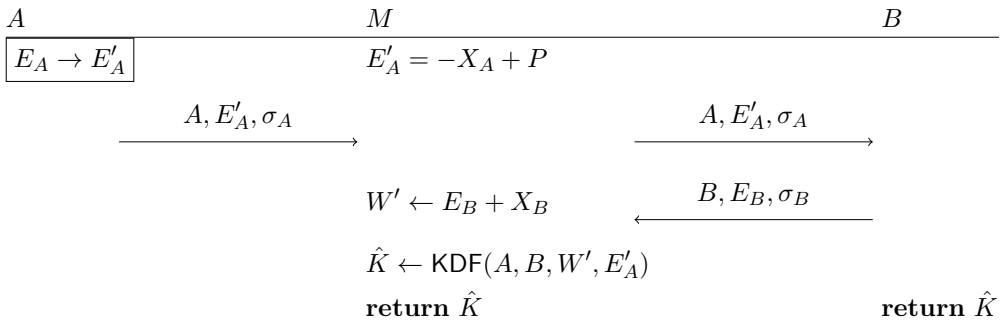

Рис. 10. Схема атаки на протокол CF с навязыванием эфемерных значений стороне A

Протокол SK6. В атаке на протокол SK6, представленный на рис. 11, нарушитель реализует угрозу KCI.

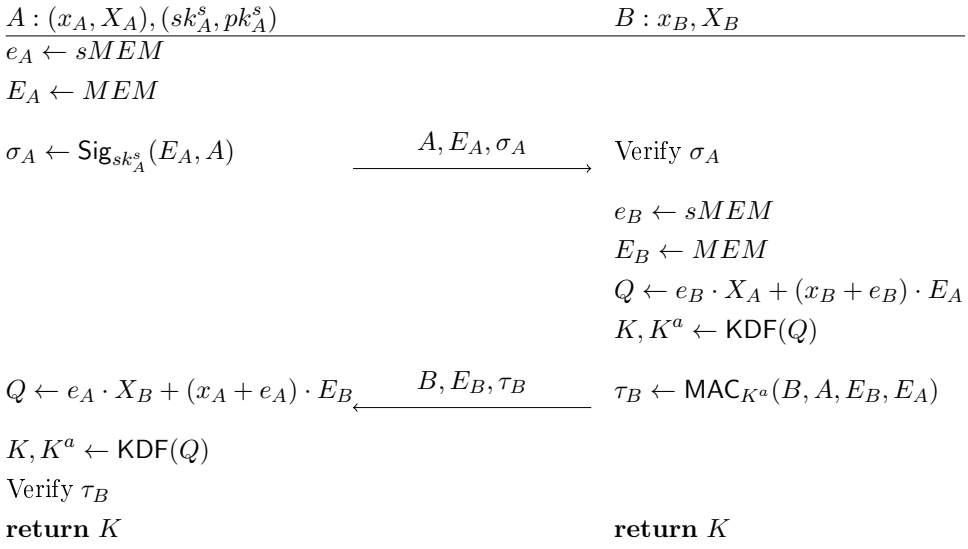

Рис. 11. Схема протокола SK6 с предвычислениями

Заметим, что протокол SK6 уязвим по отношению к угрозе KCI, если вскрыть скалярную часть долговременного ключа инициатора, то есть стороны A (без использования возможности навязывания эфемерных ключей). Поэтому рассмотрение аналогичных атак относительно стороны A с использованием соответствующей возможности нарушителя не представляет интереса. Атака с навязыванием открытых эфемерных ключей стороне A по реализации угрозы KCI относительно стороны B представлена на рис. 12. Такая атака становится возможной за счёт того, что общий ключ вычисляется по формуле $K = \mathsf{KDF}(e_B \cdot X_A + (x_B + e_B)E_A)$. Заметим, что при $E_A = -X_A$ аргумент функции KDF равен $-x_B \cdot X_A$ и не зависит от эфемерных ключей сторон.

Упомянутое свойство того, как вычисляется аргумент функции KDF, позволяет также построить атаку на протокол SK6, в результате которой нарушается свойство PFS. Для этого нарушителю понадобится навязывать открытые эфемерные значения обеим сторонам: стороне A навязывается ключ $E'_A = -X_A$, а стороне B — ключ $E'_B = -X_B$. В сеансе, где A и B используют оба этих навязанных ключа, будет выработан ключ $K = \mathsf{KDF}(-x_A \cdot X_B) = \mathsf{KDF}(-x_B \cdot X_A)$. Если после такого сеанса нарушитель

вскроет долговременный скалярный ключ какой-либо из сторон A или B, то он сможет вычислить выработанный ключ K.

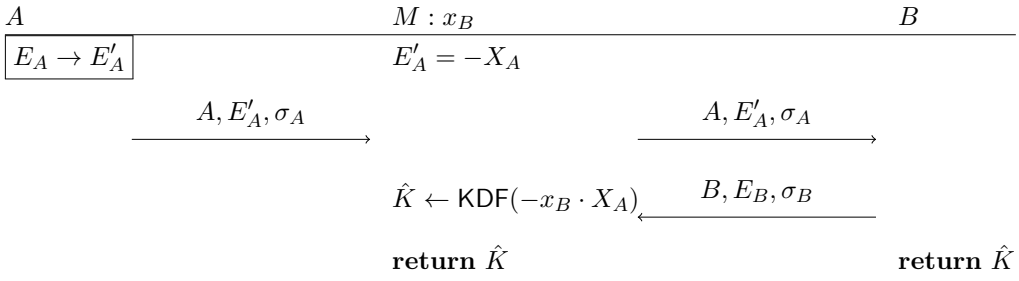

Рис. 12. Схема атаки на протокол SK6 с навязыванием эфемерных значений стороне A

Отметим, что указанное свойство позволяет нарушителю, навязывая эфемерные ключи обеим сторонам, заставить стороны A и B в любом количестве сеансов выработать одинаковый ключ, что также является уязвимостью протокола (приводит к нарушению секретности ключа при наличии у нарушителя возможности вскрывать ключи некоторых сеансов).

2.3. Протоколы, использующие KEM

Опишем атаку с навязыванием открытых эфемерных значений на протокол BKM-KK [21]. Под этим сокращённым названием мы имеем в виду протокол, который в оригинальной работе называется «Optimised KEM-based UM protocol». Схема протокола представлена на рис. 13. В данном протоколе и долговременные, и эфемерные ключи — это ключи механизма инкапсуляции ключа.

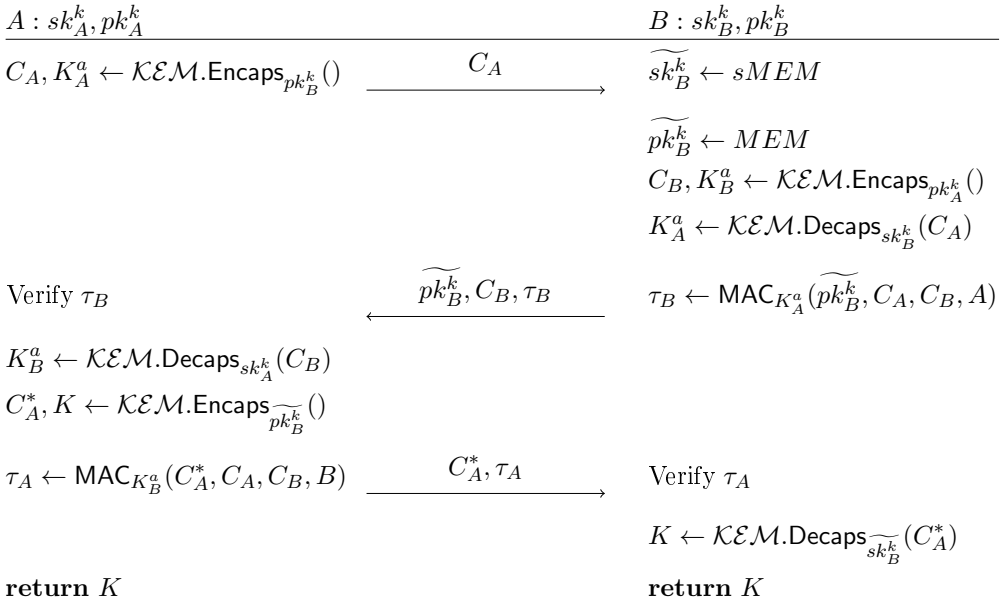

Рис. 13. Схема протокола BKM-KK с предвычислениями

Действия нарушителя при реализации атаки на протокол BKM-KK схематично приведены на рис. 14. В результате навязывания эфемерных ключей схемы KEM стороне B нарушителю удаётся выработать общий ключ со стороной A, причём A думает,

что выработала ключ со стороной B. В работе [21] приведено ещё пять протоколов, идейно схожих с протоколом BKM-KK, но отличающихся тем, какие механизмы используются для аутентификации сторон. Описанная атака применима ко всем этим протоколам.

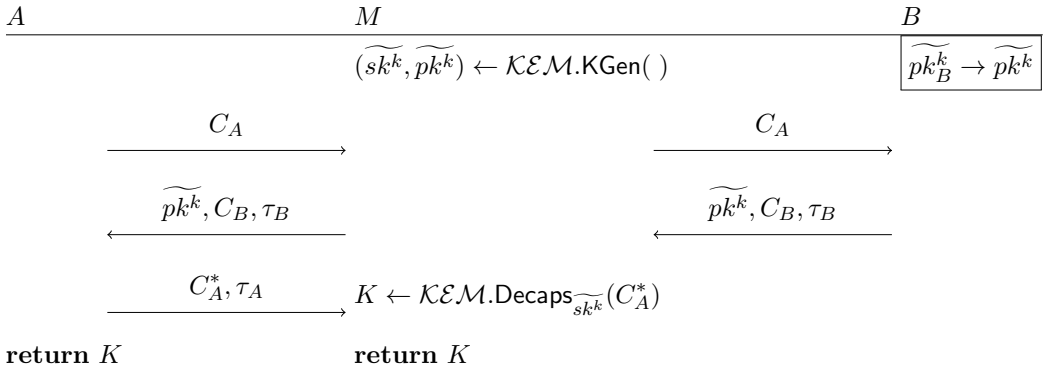

Рис. 14. Схема атаки на протокол BKM-KK с навязыванием эфемерных значений стороне A

3. Подходы к обеспечению стойкости

Приведём примеры подходов, с помощью которых можно защититься от атак, описанных в предыдущем пункте. В качестве примеров протоколов, строение которых не позволяет реализовать ни один из описанных сценариев атак, используются протоколы SIGMA-opt1 [13], «Лимонник-3» [18] и KEMTLS [22].

При построении атак на протоколы, использующие схему подписи, одной из особенностей уязвимых протоколов явилось то, что стороны вычисляли подпись от значений, не зависящих от закрытого эфемерного ключа:

— в протоколе SIG-DH+ подпись вычисляется от идентификаторов сторон и открытых эфемерных ключей;
— в протоколе SIGMA подпись вычисляется от открытых эфемерных ключей сторон;
— в протоколе SIGMA-R подпись вычисляется от своего открытого эфемерного ключа и случайности, присланной другой стороной.

При этом рассматриваемая возможность нарушителя не позволяет считать, что сторона использует соответствующий закрытый эфемерный ключ, если она прислала корректную подпись под соответствующим открытым ключом. Один из подходов к проверке соответствия закрытого и открытого ключей состоит в том, чтобы подписывать значение, которое можно вычислить только с использованием корректного закрытого ключа. Этим значением может быть имитовставка, вычисленная с помощью выработанного общего ключа (или производного от него) от данных, переданных в канале связи. Примером протокола, в котором делается именно так, является протокол SIGMA-opt1 [13] (его описание приведено на рис. 15). Причиной модификации протокола SIGMA в оригинальной работе называется экономия размера передаваемых по каналу связи данных. Однако, как показано выше, такой подход имеет преимущества и с криптографической точки зрения.

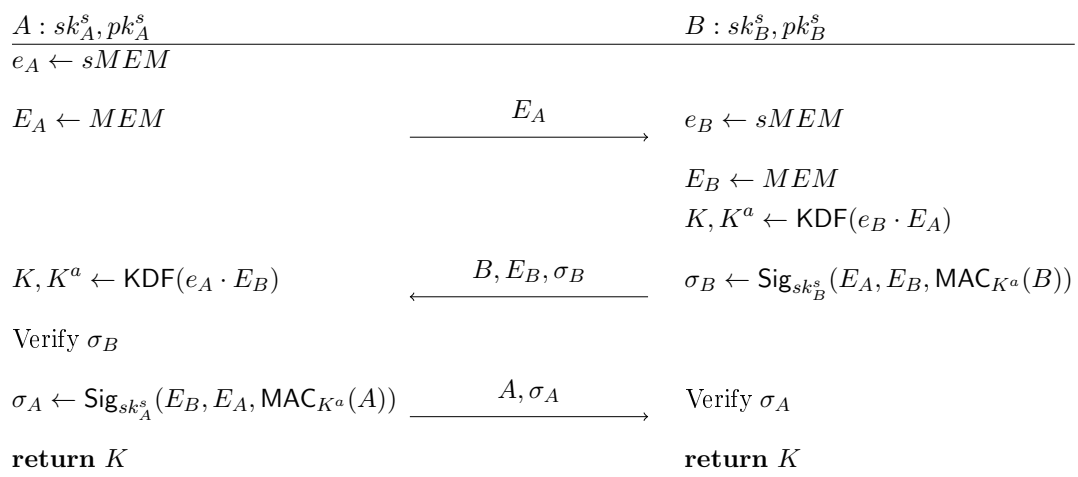

Рис. 15. Схема протокола SIGMA-opt1 с предвычислениями

Одной из причин уязвимости протоколов, использующих в качестве долговременных скалярные ключи, является то, что общий ключ вычисляется по формуле, которая позволяет манипулировать значением ключа, изменяя открытые эфемерные ключи. В протоколе CF ключ удалось сделать вычисляемым по открытым параметрам, а в протоколе SK6 — сделать его не зависящим от эфемерных ключей. Одним из подходов к тому, чтобы нарушить это свойство общего ключа, является использование при его вычислении компонент, зависящих от долговременных и эфемерных ключей как отдельных аргументов функции KDF. Недостатком такого метода является увеличение в некоторых случаях количества реализуемых операций вычисления кратной точки. Примером протокола, для которого не удаётся применить описанные сценарии атак, является другой стандартизированный в России AKE-протокол «Лимонник-3», схема которого представлена на рис. 16. Ключ в этом протоколе вычисляется по формуле (для стороны A)

$$K = \mathsf{KDF}(e_A \cdot X_B, x_A \cdot E_B, E_A, E_B).$$

Легко видеть, что навязывание, скажем, E'_B изменит лишь значение второго аргумента, но это не приносит пользы при построении атаки, так как ключи, на которых вычисляется и проверяется имитовставка, будут различны, а нарушитель это значение перевычислить не сможет, так как не знает ни одного из закрытых значений для вычисление первого аргумента функции KDF. Сценарий по реализации угрозы KCI также не реализуем, так как нарушитель, атакующий сторону A, не сможет вычислить компоненту $e_A \cdot X_B$. Что касается трудоёмкости вычислений, заметим: в отличие от протокола CF ($K = \mathsf{KDF}(A, B, (e_A + x_A)(E_B + X_B), E_A)$), в протоколе «Лимонник-3» для получения ключа требуется вычислить две кратные точки, а не одну.

В качестве протокола, использующего схемы KEM, для которого не удаётся построить атаку с навязыванием открытых эфемерных значений, приведём протокол KEMTLS [22], описанный на рис. 17. Уязвимостью протокола BKM-KK, позволившей провести на него атаку, стало то, что ключи, которые получены в результате выполнения алгоритма декапсуляции, используются сторонами исключительно для подтверждения владения закрытым ключом путём вычисления имитовставок от передаваемых в канале данных. При этом эти ключи никак не влияют на результат работы протокола, то есть на итоговый сеансовый ключ K. В протоколе KEMTLS ключи K_A и K_B, полученные в результате декапсуляции на долговременных закрытых ключах

$A : x_A, X_A$		$B : x_B, X_B$
$e_A \leftarrow sMEM$		
$E_A \leftarrow MEM$	$\xrightarrow{A, E_A}$	$e_B \leftarrow sMEM$
		$E_B \leftarrow MEM$
		$Q \leftarrow e_B \cdot X_A$
		$R \leftarrow x_B \cdot E_A$
		$K, K^a \leftarrow \mathsf{KDF}(Q, R, A, B)$
$Q \leftarrow x_A \cdot E_B$	$\xleftarrow{B, E_B, \tau_B}$	$\tau_B \leftarrow \mathsf{MAC}_{K^a}(\mathsf{c}_1, E_B, E_A, B, A)$
$R \leftarrow e_A \cdot X_B$		
$K, K^a \leftarrow \mathsf{KDF}(Q, R, A, B)$		
Verify τ_B		
$\tau_A \leftarrow \mathsf{MAC}_{K^a}(\mathsf{c}_2, E_A, E_B, A, B)$	$\xrightarrow{\tau_A}$	Verify τ_A
return K		**return** K

Рис. 16. Схема протокола «Лимонник-3» с предвычислениями

сторон, используются не только для аутентификации сторон, но и замешиваются в вырабатываемый ключ K: $K = \mathsf{KDF}(\widetilde{K_B}, K_A, K_B)$. Это не позволяет нарушителю узнать сеансовый ключ и подтвердить его значение, используя закрытый ключ схемы KEM, соответствующий навязанному одной из сторон открытому эфемерному ключу.

$A : sk_A^k, pk_A^k$		$B : sk_B^k, pk_B^k$
$\widetilde{sk_A^k} \leftarrow sMEM$		
$\widetilde{pk_A^k} \leftarrow MEM$		
$r_A \xleftarrow{U} \{0,1\}^{256}$	$\xrightarrow{\widetilde{pk_A^k}, r_A}$	$\widetilde{C_B}, \widetilde{K_B} \leftarrow \mathcal{KEM}.\mathsf{Encaps}_{\widetilde{pk_A^k}}(\,)$
$\widetilde{K_B} \leftarrow \mathcal{KEM}.\mathsf{Decaps}_{\widetilde{sk_A^k}}(\widetilde{C_B})$	$\xleftarrow{B, \widetilde{C_B}, r_B}$	$r_B \xleftarrow{U} \{0,1\}^{256}$
$C_A, K_A \leftarrow \mathcal{KEM}.\mathsf{Encaps}_{pk_B^k}(\,)$	$\xrightarrow{A, C_A}$	$K_A \leftarrow \mathcal{KEM}.\mathsf{Decaps}_{sk_B^k}(C_A)$
$K_B \leftarrow \mathcal{KEM}.\mathsf{Decaps}_{sk_A^k}(C_B)$	$\xleftarrow{C_B}$	$C_B, K_B \leftarrow \mathcal{KEM}.\mathsf{Encaps}_{pk_A^k}(\,)$
$K, K^a \leftarrow \mathsf{KDF}(\widetilde{K_B}, K_A, K_B)$		$K, K^a \leftarrow \mathsf{KDF}(\widetilde{K_B}, K_A, K_B)$
$src \leftarrow \widetilde{pk_A^k}, r_A, B, \widetilde{C_B}, r_B, A, C_A, C_B$		$src \leftarrow \widetilde{pk_A^k}, r_A, B, \widetilde{C_B}, r_B, A, C_A, C_B$
$\tau_A \leftarrow \mathsf{MAC}_{K^a}(src)$	$\xrightarrow{\tau_A}$	Verify τ_A
Verify τ_B	$\xleftarrow{\tau_B}$	$\tau_A \leftarrow \mathsf{MAC}_{K^a}(src, \tau_A)$
return K		**return** K

Рис. 17. Схема криптографического ядра протокола KEMTLS с предвычислениями (вариант двусторонней аутентификации без механизмов обеспечения анонимности)

Заключение

В работе приведены атаки на ряд AKE-протоколов (в частности, на протоколы SIGMA, SIGMA-R, STS-MAC, «Эхинацея-3» и постквантовый протокол BKM-KK), осуществление которых становится возможным при наличии у нарушителя способности навязывать участнику использование будущих открытых эфемерных значений. Приведены идеи противодействия таким атакам. Применение описанных идей позволяет защититься от приведённых в настоящей работе сценариев атак, но не гарантирует того, что нет какого-то иного успешного, но пока незамеченного сценария атаки.

Одним из способов достичь большей уверенности в стойкости AKE-протоколов, да и вообще произвольных криптосистем, является метод теоретико-сложностных сведений задачи взлома протокола к решению каких-либо математических задач, сложность которых проверена годами исследований. Однако для этого, прежде всего, необходимо разработать формальную модель безопасности таких систем. Таким образом, открытыми задачами по теме настоящей работы является разработка формальной модели безопасности, учитывающей возможность нарушителя навязывать участникам открытые эфемерные значения, а также построение сведения задачи взлома протоколов из п. 3 (или любых других протоколов) к известным трудным математическим задачам. Целесообразно также сформулировать набор синтезных принципов, следование которым позволит обеспечить защиту от атак из рассматриваемого класса.

ЛИТЕРАТУРА

1. *Алексеев Е. К.* Что плохого можно сделать, неправильно используя криптоалгоритмы? Симпозиум CTCrypt 2019. `https://ctcrypt.ru/files/files/2019/materials/29_Alekseyev.pdf`. 2019.
2. *Алексеев Е. К., Ахметзянова Л. Р., Божко А. А., Грибоедова Е. С.* Теоретическая криптография в реальных условиях. Блог компании КриптоПро. `https://cryptopro.ru/blog/2019/11/19/teoreticheskaya-cryptografiya-v-realnykh-usloviyakh`. 2020.
3. *Царегородцев К. Д., Грибоедова Е. С.* Еще раз о важности построения модели противника на примере протокола аутентификации 5G-AKA // Конференция РусКрипто'2022. `https://ruscrypto.ru/resource/archive/rc2022/files/02_tsaregorodsev_griboedova.pdf`. 2022.
4. *Degabriele J. P., Paterson K. G., and Watson G. J.* Provable security in the real world // IEEE Security & Privacy. 2011. V. 9. No. 3. P. 33–41.
5. *Алексеев Е. К., Ахметзянова Л. Р., Божко А. А. и др.* О возможностях нарушителя при атаках на некоторый класс протоколов аутентифицированной выработки общего ключа. Конференция РусКрипто'2022. `https://ruscrypto.ru/resource/archive/rc2022/files/02_alekseyev_akhmetzyanova_kutsenok_kyazhin.pdf`. 2022.
6. *Krawczyk H.* The order of encryption and authentication for protecting communications (or: How secure is SSL?) // LNCS. 2001. V. 2139. P. 310–331.
7. *Canvel B., Hiltgen A., Vaudenay S., and Vuagnoux M.* Password interception in a SSL/TLS channel // LNCS. 2003. V. 2729. P. 583–599.
8. *Kaufman C., Hoffman P., Nir Y., et al.* Internet Key Exchange Protocol Version 2 (IKEv2). RFC 7296. 2014.
9. *Sheffer Y. and Fluhrer S.* Additional Diffie — Hellman Tests for the Internet Key Exchange Protocol Version 2 (IKEv2). RFC 6989. 2013.
10. *Seye P. B. and Sarr A. P.* Enhanced modelling of authenticated key exchange security // LNCS. 2017. V. 10547. P. 36–52.

11. *Alekseev E. K., Babueva A. A., and Zazykina O. A.* AKE Zoo: 100 Two-Party Protocols (to be continued). Cryptology ePrint Archive. 2023. Paper 2023/1044.
12. *Huang H. and Cao Z.* Authenticated Key Exchange Protocols with Enhanced Freshness Properties. Cryptology ePrint Archive. 2009. Paper 2009/505.
13. *Krawczyk H.* SIGMA: The 'SIGn-and-MAc' approach to authenticated Diffie — Hellman and its use in the IKE protocols // LNCS. 2003. V. 2729. P. 400–425.
14. *Jeong I. R., Katz J., and Lee D. H.* One-round protocols for two-party authenticated key exchange // LNCS. 2004. V. 3089. P. 220–232.
15. *Jeong I. R., Katz J., and Lee D. H.* One-Round Protocols for Two-Party Authenticated Key Exchange. `https://www.cs.umd.edu/~jkatz/papers/1round_AKE.pdf`. 2008.
16. *Rescorla E.* The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446. 2018.
17. *Diffie W., Van Oorschot P. C., and Wiener M. J.* Authentication and authenticated key exchanges // Des. Codes Cryptogr. 1992. V. 2. P. 107–125.
18. Информационная технология. Криптографическая защита информации. Схемы выработки общего ключа с аутентификацией на основе открытого ключа. Р 1323565.1.004-2017. М.: Стандартинформ, 2017.
19. *Cremers C. and Feltz M.* One-round Strongly Secure Key Exchange with Perfect Forward Secrecy and Deniability. Cryptology ePrint Archive. 2011. Paper 2011/300.
20. *Song B. and Kim K.* Two-pass authenticated key agreement protocol with key confirmation // LNCS. 2000. V. 1977. P. 237–249.
21. *Boyd C., Kock B., and Millerjord L.* Modular Design of KEM-Based Authenticated Key Exchange. Cryptology ePrint Archive. 2023. Paper 2023/167.
22. *Schwabe P., Stebila D., and Wiggers T.* Post-quantum TLS without handshake signatures // Proc. 2020 ACM SIGSAC Conf. CCS'20. USA, 2020. P. 1461–1480.

REFERENCES

1. *Alekseev E. K.* Chto plokhogo mozhno sdelat', nepravil'no ispol'zuya kriptoalgoritmy? [What bad things can be done by using cryptoalgorithms incorrectly?] CTCrypt 2019 Symp. `https://ctcrypt.ru/files/files/2019/materials/29_Alekseyev.pdf`. 2019. (in Russian)
2. *Alekseev E. K., Akhmetzyanova L. R., Bozhko A. A., and Griboedova E. S.* Teoreticheskaya kriptografiya v real'nykh usloviyakh [Theoretical cryptography in the real world]. CryptoPro Blog. `https://cryptopro.ru/blog/2019/11/19/teoreticheskaya-cryptografiya-v-realnykh-usloviyakh`. 2020. (in Russian)
3. *Tsaregorodtsev K. D. and Griboedova E. S.* Yeshche raz o vazhnosti postroyeniya modeli protivnika na primere protokola autentifikatsii 5G-AKA [On the importance of making an adversary model, once again, for the 5G-AKA authentication protocol example]. RusCrypto'2022 Conf. `https://ruscrypto.ru/resource/archive/rc2022/files/02_tsaregorodsev_griboedova.pdf`. 2022. (in Russian)
4. *Degabriele J. P., Paterson K. G., and Watson G. J.* Provable security in the real world. IEEE Security & Privacy, 2011, vol. 9, no. 3, pp. 33–41.
5. *Alekseev E. K., Akhmetzyanova L. R., Bozhko A. A., et al.* O vozmozhnostyakh narushitelya pri atakakh na nekotoryy klass protokolov autentifitsirovannoy vyrabotki obshchego klyucha [On the adversary capabilities needed to attack a certain class of authenticated key establishment protocols]. RusCrypto'2022 Conf. `https://ruscrypto.ru/resource/archive/rc2022/files/02_alekseyev_akhmetzyanova_kutsenok_kyazhin.pdf`. 2022. (in Russian)
6. *Krawczyk H.* The order of encryption and authentication for protecting communications (or: How secure is SSL?). LNCS, 2001, vol. 2139, pp. 310–331.

7. *Canvel B., Hiltgen A., Vaudenay S., and Vuagnoux M.* Password interception in a SSL/TLS channel. LNCS, 2003, vol. 2729, pp. 583–599.
8. *Kaufman C., Hoffman P., Nir Y., et al.* Internet Key Exchange Protocol Version 2 (IKEv2). RFC 7296, 2014.
9. *Sheffer Y. and Fluhrer S.* Additional Diffie — Hellman Tests for the Internet Key Exchange Protocol Version 2 (IKEv2). RFC 6989, 2013.
10. *Seye P. B. and Sarr A. P.* Enhanced modelling of authenticated key exchange security. LNCS, 2017, vol. 10547, pp. 36–52.
11. *Alekseev E. K., Babueva A. A., and Zazykina O. A.* AKE Zoo: 100 Two-Party Protocols (to be continued). Cryptology ePrint Archive, 2023, Paper 2023/1044.
12. *Huang H. and Cao Z.* Authenticated Key Exchange Protocols with Enhanced Freshness Properties. Cryptology ePrint Archive, 2009, Paper 2009/505.
13. *Krawczyk H.* SIGMA: The 'SIGn-and-MAc' approach to authenticated Diffie — Hellman and its use in the IKE protocols. LNCS, 2003, vol. 2729, pp. 400–425.
14. *Jeong I. R., Katz J., and Lee D. H.* One-round protocols for two-party authenticated key exchange. LNCS, 2004, vol. 3089, pp. 220–232.
15. *Jeong I. R., Katz J., and Lee D. H.* One-Round Protocols for Two-Party Authenticated Key Exchange. `https://www.cs.umd.edu/~jkatz/papers/1round_AKE.pdf`. 2008.
16. *Rescorla E.* The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, 2018.
17. *Diffie W., Van Oorschot P. C., and Wiener M. J.* Authentication and authenticated key exchanges. Des. Codes Cryptogr., 1992, vol. 2, pp. 107–125.
18. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Skhemy vyrabotki obshchego klyucha s autentifikatsiyey na osnove otkrytogo klyucha [Information Technology. Information Cryptographic Protection. Public Key Based on the Authenticated Key Agreement Schemes]. R 1323565.1.004-2017. Moscow, Standartinform Publ., 2017. (in Russian)
19. *Cremers C. and Feltz M.* One-round Strongly Secure Key Exchange with Perfect Forward Secrecy and Deniability. Cryptology ePrint Archive, 2011, Paper 2011/300.
20. *Song B. and Kim K.* Two-pass authenticated key agreement protocol with key confirmation. LNCS, 2000, vol. 1977, pp. 237–249.
21. *Boyd C., Kock B., and Millerjord L.* Modular Design of KEM-Based Authenticated Key Exchange. Cryptology ePrint Archive, 2023, Paper 2023/167.
22. *Schwabe P., Stebila D., and Wiggers T.* Post-quantum TLS without handshake signatures. Proc. 2020 ACM SIGSAC Conf. CCS'20, USA, 2020, pp. 1461–1480.