Реализация криптопротокола распределения ключей Менезиса-Кью-Ванстоуна

Б. Н. Воронков, email:vrnkv@mail.ru
А. А. Школин, email:mr.shkolin.1998@mail.ru

Воронежский государственный университет

Аннотация. *Рассмотрены этапы двадцатилетнего перехода от первого, классического протокола Диффи-Хеллмана распределения закрытых ключей по незащищённым каналам связи к стандартизированному протоколу Менезиса-Кью-Ванстоуна, моделирование и анализ работы данного криптографического протокола.*

Ключевые слова: *Асимметричные криптографические системы, криптографические протоколы, дискретное логарифмирование.*

Введение

Предложенный в 1976 году двумя американскими математиками У. Диффи и М. Хеллманом [1], первый в истории криптографии протокол распределения ключей по незащищённым каналам связи, подтвердил, за более чем сорок лет применения, свою эффективность и надёжность. Безопасность данного криптографического протокола обусловлена вычислительной трудностью нахождения дискретных логарифмов в конечном поле большого объёма. Протокол Диффи-Хеллмана положил начало двухключевой (асимметричной) криптографии и, фактически, решил многовековую проблему функционирования симметричных (одноключевых) криптосистем, в которых две стороны обмена сообщениями должны доверять друг другу и пользоваться единым, секретным ключом при шифровании.

В последующем было предложено ещё несколько протоколов [2]. Так, в 1978 году был опубликован транспортный протокол Нидхема-Шрёдера, который требует предварительной аутентификации открытых ключей абонентов A и B. В 1980 году – бесключевой трёхэтапный протокол Ади Шамира, тоже подверженный атаке «Человек посередине». В 1988 году появился первый вариант стандарта X.509 для инфраструктуры открытых ключей и инфраструктуры управления привилегиями. Данный протокол включал в себя не только передачу подключей абонентов, из которых и формировался общий секретный ключ, но и аутентификацию сторон с использованием цифровой

© Воронков Б. Н., Школин А. А., 2021

подписи. Кроме того, на одном из шагов протокола абонент А должен был подтверждать получение сообщения от В при помощи отправления подписанного сообщения. Использование меток времени противодействовало атакам с повторным использованием сообщений или с блокированием канала и отправлением искажённых (подделанных) сообщений. Принятию искажённых сообщений противодействовало также наличие цифровой подписи.

Наконец, сочетание нескольких недостатков предыдущих протоколов было устранено в 1995 году применением протокола Менезиса-Кью-Ванстоуна (MQV-протокола) [3], стандартизованного в 2000-м году [4].

1. Протокол Диффи-Хеллмана

Суть протокола состоит в следующем. Два абонента А и В хотят получить общий секретный ключ для использования, в дальнейшем, симметричной криптосистемы. Для этого А и В согласованно выбирают два больших целых числа (порядка двухсот или более десятичных разрядов): p и α . При этом

1. Пусть p – большое простое число ($p \sim 10^{300}$), α – примитивный корень (элемент) простого поля Галуа; $\alpha \in Z_p^*$, $1 < \alpha \le p - 1$; $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{p-1} \equiv 1 (\mathrm{mod}\ p) \in Z_p^*$. α и p – общедоступны.

2. K_A и K_B –закрытые ключи пользователей А и В. K_A и K_B – большие, случайные целые числа.

3. А: $y_A = \alpha^{K_A} \bmod p$; В: $y_B = \alpha^{K_B} \bmod p$; y_A и y_B – открытые ключи пользователей.

4. А и В обмениваются открытыми ключами по незащищённому каналу.

5. А и В вычисляют общий секретный ключ:

А: $K_S = (y_B)^{K_A} \bmod p = (\alpha^{K_B})^{K_A} \bmod p$;

В: $\hat{K}_S = (y_A)^{K_B} \bmod p = (\alpha^{K_A})^{K_B} \bmod p$;

$K_S = \hat{K}_S$, так как $(\alpha^{K_B})^{K_A} = (\alpha^{K_A})^{K_B}$. Теперь общий секретный ключ K_S можно использовать для обмена шифрованными данными на основе симметричной криптосистемы. Например, воспользовавшись межгосударственным стандартом ГОСТ Р 34.12-2018 [5]. Таким образом, для получения общего секретного ключа абонентам А и В необходимо выполнить всего лишь несколько несложных

вычислений в модулярной арифметике, а вот злоумышленнику (даже при известных или перехваченных числах p, α, y_A, y_B) потребуется решить вычислительно трудную задачу дискретного логарифмирования. Важно: p – должно быть большим простым числом (порядка 10^{300}); $\frac{p-1}{2}$ – тоже должно быть простым числом; α – примитивный корень по модулю p (в принципе достаточно, чтобы число α генерировало большую подгруппу мультипликативной группы по модулю p).

Простой протокол обмена ключами Диффи-Хеллмана отлично справляется с пассивной атакой, но, к сожалению, не обеспечивает ни одного из основных свойств протоколов распределения ключей: ни аутентификацию параметров, ни подтверждение ключа, ни аутентификацию участников протокола [6]. Активный противник может построить атаку на протокол методом включения в канал (атака "Человек посередине"). В итоге он сможет контролировать весь обмен данными между участниками. При этом они не смогут обнаружить подмену данных и будут уверены, что связываются непосредственно друг с другом.

Данный недостаток протокола Диффи-Хеллмана может быть преодолён осуществлением взаимной аутентификации, например, с использованием электронной подписи. Если у абонента B имеется открытый ключ электронной подписи абонента A, и он уверен, что это действительно ключ адресата A, то для защиты от атаки "Человек посередине" абоненту A достаточно подписать своим закрытым ключом электронной подписи число y_A. Теперь злоумышленник не сможет выдать себя за абонента A, так как не сможет подделать его электронную подпись. Однако, такое преодоление атаки "Человек посередине" приводит к необходимости увеличения размера передаваемых сообщений, зачастую в несколько раз.

2. Протокол Менезиса-Кью-Ванстоуна

Протокол MQV состоит в следующем. Абоненты A и B имеют каждый свою ключевую пару, состоящую из открытых и закрытых ключей. A: ($y_A = \alpha^{K_A} \bmod p$; K_A) и B: ($y_B = \alpha^{K_B} \bmod p$; K_B). Абоненту B известен открытый ключ y_A, а абоненту A известен открытый ключ y_B. Далее A и B генерируют сеансовую пару ключей A:

($C = \alpha^{\gamma} \bmod p$; γ) и B: ($D = \alpha^{\delta} \bmod p$; δ). Затем происходит обмен открытыми сеансовыми ключами, как в классическом протоколе Диффи-Хеллмана: A пересылает B открытый ключ C, а B пересылает A открытый ключ D. Теперь A знает: y_A; y_B; C; D; K_A; γ. Абоненту B известны: y_A; y_B; C; D; K_B; δ.

Для получения общего секретного ключа K_S абонент A выбирает число λ, равное размеру сообщения в битах, делённому на два (для протокола MQV на эллиптических кривых длина сообщения равна 160 бит, следовательно λ=80). Далее абонент A:

Задаёт i=C.

Находит $S_A = (i \pmod{2^{\lambda}}) + 2^{\lambda}$.

Задаёт j=D.

Вычисляет $T_A = (j \pmod{2^{\lambda}}) + 2^{\lambda}$.

Находит $h_A = \gamma + S_A \cdot K_A$.

Вычисляет $P_A = (D \cdot y_B^{T_A})^{h_A} \bmod p$.

Абонент B проделывает те же действия, но со своими закрытыми ключами:

Задаёт i=D.

Находит $S_B = (i \pmod{2^{\lambda}}) + 2^{\lambda}$.

Задаёт j=C.

Вычисляет $T_B = (j \pmod{2^{\lambda}}) + 2^{\lambda}$.

Находит $h_B = \delta + S_B \cdot K_B$.

Вычисляет $P_B = (C \cdot y_A^{T_B})^{h_B} \bmod p$.

Найденные числа $P_A = P_B = K_S$ и являются общим секретным ключом.

Действительно, используя дискретное логарифмирование, или по другому, нахождение индекса по модулю p при основании α, получим.

$ind_{\alpha}(P_A) = ind_{\alpha}((D \cdot (y_B)^{T_A})^{h_A}) = (\delta + K_B \cdot T_A) \cdot h_A$, так как $D = \alpha^{\delta} \bmod p$ и $y_B = \alpha^{K_B} \bmod p$. Далее преобразуем

$(\delta + K_B \cdot T_A) \cdot h_A = \delta \cdot (\gamma + S_A \cdot K_A) + K_B \cdot T_A \cdot (\gamma + S_A \cdot K_A)$,

так как $h_A = \gamma + S_A \cdot K_A$.

$$\delta \cdot (\gamma + S_A \cdot K_A) + K_B \cdot T_A \cdot (\gamma + S_A \cdot K_A) = \delta \cdot (\gamma + T_B \cdot K_A) + K_B \cdot S_B (\gamma + T_B \cdot K_A)$$

,

так как $S_A = T_B$ *и* $T_A = S_B$.

$$\delta \cdot (\gamma + T_B \cdot K_A) + K_B \cdot S_B \cdot (\gamma + T_B \cdot K_A) = \gamma \cdot (\delta + S_B \cdot K_B) + K_A \cdot T_B (\delta + S_B \cdot K_B) =$$

$= (\gamma + K_A \cdot T_B)^{h_B}$, *так как* $h_B = \delta + S_B \cdot K_B$.

$$(\gamma + K_A \cdot T_B)^{h_B} = ind_\alpha ((C \cdot y_A^{T_B})^{h_B} = ind_\alpha (P_B) = ind_\alpha (P_A).$$

Таким образом, корректность протокола доказана.

Следует заметить, что в преобразованиях используются закрытые ключи как абонента А, так и В. Следовательно, любой пользователь протокола может быть уверен, что кроме того абонента, с которым он хочет установить связь, получить общий секретный ключ не удастся никому.

Несмотря на некоторую сложность представления протокола, в скорости MQV-протокол ничего не теряет по сравнению со схемой, использующей электронную подпись. Дело в том, что в обоих случаях используются одни и те же операции возведения в степень по модулю простого числа.

Преимущества протокола MQV:

Устойчивость к атаке «Человек посередине».

Небольшой размер сообщения.

Удобная реализация протокола, не требующая от пользователя электронной подписи под каждым сообщением.

В сравнении с методом RSA, формирование общего закрытого ключа происходит в сотни раз быстрее. В криптосистеме RSA генерация новых закрытых и открытых ключей основана на генерации новых, больших простых чисел, что занимает много времени.

В соответствии с описанием MQV-протокола была реализована программа, позволяющая провести анализ данного протокола.

3. Описание программы

Общие сведения

MQV – название программы, её обозначение –  . Алгоритм получения общего секретного ключа реализован в интегрированной среде разработки программного обеспечения Microsoft Visual Studio 2010 на языке программирования C# (C Sharp). Для эксплуатации этой программы не требуются специальные программы систем обработки информации и программных документов, но для внесения изменений в код программы необходима среда разработки Microsoft Visual Studio.

Функциональное назначение

Программа позволяет получить общий секретный ключ двум пользователям, путём использования долговременного и сеансового ключей. Так же, есть возможность пронаблюдать, какие ключи используются в примере, какие промежуточные значения были получены, результат работы алгоритма и сам общий секретный ключ.

Описание логической структуры

Алгоритм программы:

1) переменной p присваивается значение большого простого числа;

2) переменной α присваивается значение примитивного корня;

3) генерируем переменные alfa, betta, которые являются долговременными секретными ключами;

4) вычисляем переменные A, B, которые являются открытыми долговременными ключами;

5) генерируем переменные gamma, delta, которые являются сеансовыми секретными ключами;

6) вычисляем переменные C, D, которые являются открытыми сеансовыми ключами; переменной l присваивается значение 2^{80};

7) вычислительные действия пользователя A:

а) переменной i присваивается значение C; б) вычисляется значение переменной Sa; в) переменной j присваивается значение D; г) вычисляется значение переменной Ta; д) вычисляется значение переменной ha; е) вычисляется значение переменной Pa;

8) вычислительные действия пользователя B:

а) переменной i присваивается значение D; б) вычисляется значение переменной Sb; в) переменной j присваивается значение C; г) вычисляется значение переменной Tb; д) вычисляется значение переменной hb; е) вычисляется значение переменной Pb.

Далее все результаты выводятся в окне приложения, происходит проверка полученных значений Pa и Pb, и выводится результат, совпал ли общий секретный ключ для пользователей.

Подключение библиотеки System.Numerics позволяет использовать тип BigInteger – очень длинные числа, и необходимые методы.

Метод ModPow – выполняет модульное деление числа, возведённого в степень другого числа.

Класс Random предоставляет генератор случайных чисел, с помощью которого можно генерировать натуральное число из нужного интервала.

Модуль DateTime – используется для замера времени работы программы.

Операция % – нахождение остатка от деления двух чисел.

Используемые технические средства

Программа будет корректно выполняться на компьютерах не ниже Intel Pentium 100, с размером оперативной памяти более 10 Мб, и объёмом жёсткого диска более 512 Мб.

Вызов и загрузка

Программа вызывается исполняемым файлом MQV.exe, размер которого 18 килобайт. Программа использует для работы несколько мегабайт оперативной памяти.

Входные данные

Переменная p – переменная типа BigInteger, которая хранит значение очень большого простого числа p.

Переменная g – переменная типа integer, для хранения значения примитивного корня.

Переменные A, B, alfa, betta, C, D, gamma, delta – переменные типа BigInteger, которые хранят значения ключей.

Остальные переменные высчитываются по необходимым формулам и выводятся на экране в соответствующих полях.

Выходные данные

Итогом алгоритма является общий секретный ключ для двух пользователей, полученный с помощью долговременных и сеансовых ключей. Пример работы алгоритма и результат проверки совпадения ключей приведены на рисунке 1.

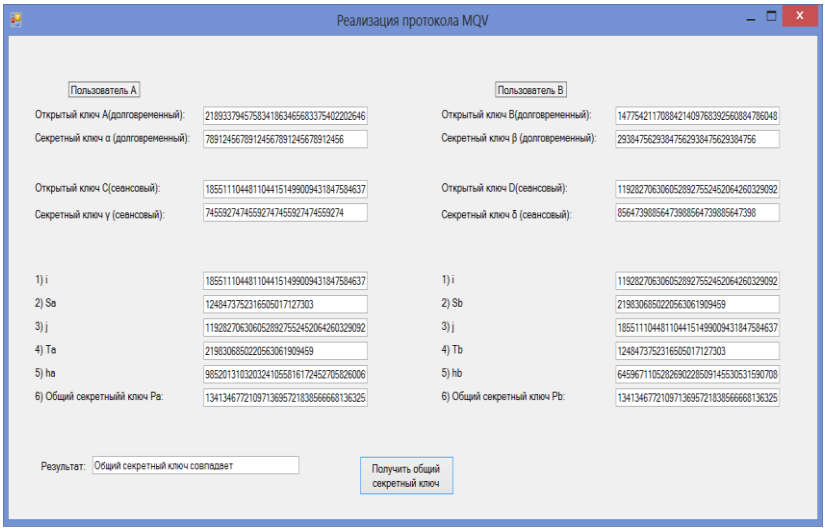

Рис. 1. Пример работы протокола MQV

Вычислительный эксперимент

Итоговая операция вычисления общего секретного ключа выполняется по модулю p, а это значит, что длина ключа будет примерно равна длине числа p. Попробуем взять разную длину числа p и пронаблюдаем, как изменится время вычисления общего секретного ключа в алгоритме.

Возьмём длину числа p, равную 32, 64, 96, 128, 160, 192, 224 и 256 десятичных цифр. На рисунке 2 приведён график, на котором отображено время работы программы в зависимости от длины числа p.

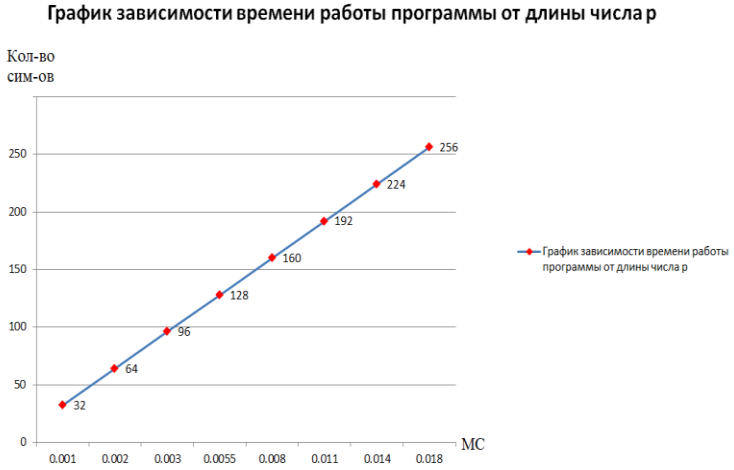

Рис. 1. График зависимости времени работы программы от длины ключа

Заключение

Таким образом, в работе рассмотрены этапы двадцатилетнего перехода от первого, классического протокола Диффи-Хеллмана распределения секретных ключей по незащищённым каналам связи к стандартизированному протоколу Менезиса-Кью-Ванстоуна. Кратко описаны достоинства и недостатки, разработанных в этот период протоколов. Реализовано компьютерное моделирование протокола MQV. Проведённый вычислительный эксперимент и выявленный линейный характер зависимости времени работы компьютерной программы от длины ключа подтверждают теоретические предположения и позволяют обоснованно выбирать значения параметров протокола MQV, а также возможность, на основе данной программы, создавать новые

приложения для защищённой связи с использованием современных стандартов блочного и поточного шифрования.

Список литературы

1. Diffie B. W. New Directions in Cryptography / B. W. Diffie, M. E. Hellman // IEEE Transactions on Information Theory, vol. IT-22, Nov. 1976. – Pp. 644 – 654.
2. Ниссенбаум О. В. Криптографические протоколы: Учебное пособие / О. В. Ниссенбаум. – Тюмень: Изд-во Тюменского гос. ун-та, 2007. – 112 с.
3. Menezes A. Some New Key Agreement Protocols Providing Implicit Authentication / A. Menezes, M. Qu, S. Vanstone // workshop record, 2nd Workshop Selected Areas in Cryptography (SAC-95), Ottawa, Canada, May 1995. – Pp. 22 – 32.
4. Standard IEEE P1363-2000. – URL: https://perso.telecom-paristech.fr/guilley/recherche/cryptoprocesseurs/ieee/00891000.pdf (дата обращения 5.01.2021)
5. ГОСТ Р 34.12-2018. Информационная технология. Криптографическая защита информации. Блочные шифры. М.: Стандартинформ, 2018. – 17 с.
6. Открытое распределение ключей.– (URL: http://cryptowiki.net/index.php?title=Открытое_распределение_ключей) (дата обращения: 5.01.2021)