УДК 681.3:519.872

А.В. Анисимов, А.В. Зубенко

ДИНАМИЧЕСКИЕ КОАЛИЦИИ — НОВАЯ ПАРАДИГМА В ОБЛАСТИ РАСПРЕДЕЛЕННЫХ КОМПЬЮТЕРНО-КОММУНИКАЦИОННЫХ СИСТЕМ. Ч. 1. ОСНОВНЫЕ АЛГОРИТМЫ КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ

Дается обзор нового направления в области распределенных вычислительных систем, связанного с защитой информации в коалиционных групповых объединениях. Рассматриваются вопросы определения динамической коалиции, а также возможных типов динамических коалиционных сред. Приведены основные алгоритмы защиты коммуникационных обменов в коалиционных группах.

Введение

В современных условиях при решении большинства сложных практических задач возникает острая необходимость использования распределенных информационных ресурсов. Этому способствуют насыщение парка вычислительных систем и значительный прогресс в коммуникационных технологиях. На первый план выдвигается новая структурная организация информационных взаимодействий, связанная с объединением удаленных пользователей в коалиционные группы, состав которых меняется динамически. Члены группы имеют заданный приоритет при использовании информационных ресурсов и выполняют целевые функции, определенные внутри группы. При реализации коалиционных структур возникает ряд новых задач, решение которых как на теоретическом, так и на практическом уровне задает развитие информационной коалиционной технологии. Так, например, при вступлении пользователя в коалиционную группу или при его выходе из такой группы необходимо переопределять права доступа всех членов группы к информационным ресурсам. Это требует создания специальных протоколов изменения ключей шифрования для коалиционной группы.

Коалиционная тематика имеет важное значение для всех областей человеческой деятельности, выдвигающих повышенные условия к защите распространяемой информации. Коалиционная система способна предоставить среду для взаимодействия военных союзников, крупных концернов, имеющих общие интересы, а также групп партнеров в сферах малого бизнеса и научных разработок.

Коалиционная технология получила свое непосредственное развитие в рамках изучения вопросов построения моделей, защищенных групповых приложений, изучение которых началось в середине 90-х гг. Ссылки на первые работы по данной тематике можно найти в [1]. Тематика постоянно развивалась в связи с масштабным распространением глобальных сетей, их слиянием в Интернет и распространением влияния удаленного сетевого сотрудничества на все виды человеческой деятельности — от коммерческих до глобальных военных. В настоящее время коалиционная технология стремительно развивается и становится доминирующим фактором в области построения распределенных систем партнерского взаимодействия.

Динамические коалиции — это, в первую очередь, разновидность сложных распределенных вы-

числительных систем. Сложность системы рассматривается под углом сложности ее инфрастуктуры. Чем больше элементов в распределенной системе и чем сложнее их иерархия, тем сложнее процесс ее управления. Методы управления коалицией должны быть эффективными независимо от сложности внутренней структуры ее элементов — это одно из главных условий, которым должна отвечать успешная реализация коалиционной схемы. Элементами коалиции, как правило, выступают целые программные комплексы, существенно удаленные с точки зрения географической локализации, но требующие надежного метода обмена данными в реальном времени.

Роль распределенных вычислительных систем в целом сегодня возросла до воистину грандиозных масштабов, и количество повсеместно применяемых подобных систем, обеспечивающих всевозможные информационные нужды, продолжает неуклонно расти. Примеров подобных систем можно привести множество. Среди них — элементарное пользование потребительскими онлайн-услугами, а также более сложные варианты, такие, как участие в многосторонних онлайн-аудио и видеоконференциях, выполнение сложных финансовых операций и пр. Все это лишь малая часть всеобъемлющего и стремительно развивающего глобального процесса проникновения информационных технологий в большинство сфер человеческой деятельности. Какова же роль коалиций во всех этих сложных процессах, а также в чем собственно отличие коалиционных схем от существующих схем и стандартов?

Разработка сложной распределенной информационной системы — это сама по себе непростая и ресурсоемкая задача, требующая немалых финансовых и производственных затрат. Было приложено немало усилий IT-специалистов в области разработки интегрированных аппаратно-программных комплексов, способных упростить решение задач, существующих в данной области. Поэтому и существуют готовые к использованию сложные производительные кластерные платформы, параллельные вычислительные алгоритмы, множество семейств протоколов распределенного обмена данными, методы балансировки нагрузки на распределенную систему, технологии использования удаленных объектов и прочие не менее важные достижения. Исследования в данных направлениях нельзя считать законченными, так как реальность диктует все новые и новые задачи. Тем не менее на сегодня существует немало дееспособных методов реализации распределенных систем, реализующих различные информационные обмены [2]. В связи с этим изучение коалиций в большей мере фокусируется на проблеме защиты ресурсов, разделяемых в рамках коалиции, от несанкционированного доступа. Повышенные требования к защите ресурсов являются наиболее существенной характеристикой коалиции, отличающей ее от иных распределенных систем партнерского взаимодействия.

Приведем список наиболее актуальных проблем, связанных с построением систем безопасности для различных коалиционных сред.

- Изучение методов групповой криптографии.
- Построение стойких к отказам и атакам распределенных систем сертификации открытых ключей.
- Создание протоколов общего доступа к совместно обладаемым партнерами по коалиции ресурсам, не допускающих вариант эксклюзивного доступа к ним одним из партнеров.

• Нахождение методов эффективного, в условиях реального времени, обеспечения свойства динамичности коалиционной среды. Данное свойство позволяет осуществить динамическое управление составом партнеров и своевременно реагировать на изменение уровней доверительных отношений между ними, таким образом реализуя динамический контроль доступа к разделяемым ресурсам.

• Изучение методов защиты мобильного кода от влияния среды выполнения и обратной защиты среды выполнения от действий злоумышленного кода.

1. Динамическая коалиционная среда

Приведем неформальное определение коалиционного окружения.

Под ***динамической коалиционной средой (динамической коалицией)*** *будем понимать распределенную систему комплексов взаимодействующих приложений, целью функционирования которой является предоставление использующим ее партнерам средств, позволяющих разделять ресурсы в реальном времени и кооперировать свои возможности для достижения совместных целей при помощи контролируемых, защищенных и поддающихся учету методов.*

Следует отметить, что формального подхода к определению всех составляющих коалиционного окружения, позволяющих классифицировать ту или иную распределенную систему как динамическую коалицию, на данный момент не существует. В [3-5] и ряде других, рассматриваемых детально во второй части данной статьи, предложены конкретные решения, имеющие как математическую основу, так и практические реализации отдельно взятых прототипов коалиционных окружений. При этом разработчики данных моделей полагаются на весьма обобщенное определение динамической коалиции, схожее с приведенным выше.

Подчеркнем следующий факт: ошибкой будет абстрагирование любой распределенной системы, обладающей элементами защиты разделяемых ресурсов от несанкционированного доступа, до понятия коалиции. Подавляющее большинство современных распределенных систем не удовлетворяет четкому ряду условий, определенных ниже, необходимых для отнесения ее к разряду коалиций. Приведем несколько примеров. Двунаправленное SSL/TLS соединение между клиентом и сервером не является коалицией. Множество подобных соединений клиентов и кластера серверов также не является коалицией. IPSec-туннель или иная система класса VPN, соединяющая удаленные отделения единой корпорации, также не удовлетворяют условиям для отнесения к классу коалиций. Данные широко применяемые схемы защиты распределенных систем могут только являться составными частями сложных коалиционных схем. Коалицию характеризует динамизм изменения состава партнеров, набора взаимодействующих приложений, целей, преследуемых участниками, а также наличие механизмов надежного контроля, анализа и адекватной и своевременной реакции на события, происходящие в данном динамическом окружении.

Как результат анализа существующих разработок в области коалиционных систем можно выделить пять достаточных свойств, условие выполнения которых позволяет относить распределенную систему к классу динамических коалиций.

1. Динамическое изменение состава участников коалиции. Возможность объединения или распада коалиций с целью создания

новых, независимых и защищенных от влияния прежних партнеров коалиционных окружений.

2. Динамическое изменение целей и задач, преследуемых участниками коалиции, возможно, вследствие изменения уровней их взаимного доверия по отношению друг к другу.

3. Механизм реализации трех основных задач криптографической защиты информационных обменов (аутентификация, целостность, конфиденциальность) в рамках данного распределенного группового взаимодействия.

4. Наличие сетевой инфраструктуры и технологического каркаса, способных обеспечить надежное выполнение определенных выше свойств в условиях реального времени.

5. Сотрудничество в рамках коалиции должно быть защищено от каких-либо внешних воздействий, однако его непосредственные участники не лишены возможности пользования внешними информационными услугами.

Этим пяти условиям, на наш взгляд, должна отвечать архитектура успешной коалиционной модели в общем случае. Важность п. 4 является особо критичной — невыполнение данного условия лишает любую модель практической значимости. При определенных условиях допустимо расширение или сужение данного набора свойств, присущих защищенной системе партнерского взаимодействия, однако, на наш взгляд, в случае невыполнения хотя бы одного из свойств классифицировать данную систему как коалицию было бы неверным.

Приведем трехуровневую архитектуру коалиционной модели, опираясь на которую можно спроектировать произвольную коалиционную модель.

Будем рассматривать коалицию как интегрированную систему, структура которой состоит из трех взаимозависимых компонентов:

1. *Групповая Стратегия (ГС)* — логическое ядро создаваемой коалиции. Данный компонент отвечает за построение логических механизмов управления составом и ресурсами коалиции.

2. *Уровень защиты (УЗ)*, включающий в себя протоколы распределения и отзыва ключей, сертификатов и других криптографических параметров, наличие которых зависит от конкретной политики безопасности, утвержденной на этапе проектирования.

3. *Групповая Коммуникационная Инфраструктура (ГКИ)* — технологическая основа коалиции. С одной стороны, охватывает вопросы построения надежной коммуникационной платформы, отвечающей потребностям партнеров по обмену данными, что включает в себя в общем случае наличие надежного метода целевого широковещания, а также, с другой стороны, вопросы выбора и интеграции необходимых технологических средств, способных создать платформу для коалиционного взаимодействия.

Технология надежного целевого широковещания (reliable multicast), указанная в п. 1, служит для передачи данных от произвольного отправителя целевой группе настроенных на ее прием получателей, с учетом гарантии доставки всей целевой группе. Данная технология зачастую необходима при реализации групповых коммуникаций, лежащих в основе коалиционных взаимодействий.

Порядок, в котором перечислены компоненты, отражает условную уровневую архитектуру коалиции. Первым шагом в решении данного вопроса должно быть четкое определение второго компонента коалиционной схемы — групповой стратегии, являющейся ее логическим ядром. Перечислим

ряд приоритетных вопросов, охватываемых групповой стратегией.

- правила и условия формирования коалиции, изменения состава ее участников, условий распада;
- определения классов приложений, взаимодействие которых предусмотрено в данной коалиционной среде;
- определения методов мониторинга и контроля за происходящими в рамках коалиции процессами;

Перечисленные выше пункты позволяют получить общее представление о групповой стратегии. Более строгое определение компонентов групповой стратегии представляет собой первостепенное и высокоприоритетное направление дальнейших исследований.

После четкого определения групповой стратегии может быть определена политика безопасности данной распределенной вычислительной системы — второй компонент.

Завершает проектирование коалиции корректный подбор технологических средств, определяющих третий компонент.

Наличие подобной уровневой архитектуры коалиционной модели значительно упрощает задачу анализа существующих сегодня достижений в рамках коалиционной тематики, выявлять их критические элементы, а также необходимые пути их модификации, а также определять приоритетные направления дальнейших исследований. В то же время она должна послужить отправной точкой для разработки новых коалиционных моделей.

2. Основные алгоритмы групповой криптографии

В настоящее время все известные алгоритмы криптографической защиты обменов информацией внутри группы основываются на процедурах формирования единого секретного ключа для всех участников группы. Кроме этого защищенное групповое взаимодействие требует не только установления начального ключа, но и его трансформации вследствие динамического изменения состава группы, к которому могут вести следующие четыре типа событий.

- **Добавление.** Новый элемент включается в состав группы.
- **Удаление.** Текущий элемент покидает группу.
- **Слияние.** Совокупность новых элементов присоединяется к группе.
- **Отделение.** Совокупность новых элементов выводится из группы.

События **добавления** и **удаления** могут рассматриваться как частные случаи событий **слияния** и **отделения.** Последние два типа событий могут возникать непроизвольно в нестойких к отказам глобальных сетевых окружениях. **Отделение** может быть следствием отказа оборудования и каналов передачи данных, что вызывает недоступность целой группы участников коммуникационного процесса для остальных его элементов. После восстановления коммуникационного канала вся ранее отключенная совокупность элементов может возобновить групповое сотрудничество, проведя операцию **слияния.**

Существует ряд криптографических свойств, которым должны удовлетворять протоколы управления групповыми ключами.

1. Секретность группового ключа (Group Key Secrecy), гарантирующая практическую невозможность вычисления группового ключа пассивным противником.
2. Прямая секретность (Forward Secrecy), гарантирующая невозможность вычисления пассивным противником последующих ключей на основании известного последовательного набора предшествующих ключей.
3. Обратная секретность (Backward Secrecy) — обратная прямой характеристика, гарантирующая невозможность вычисления предшествующих ключей на основании известного набора последующих.
4. Независимость ключей — наиболее строгая характеристик в секретности, гарантирующая тот факт, что пассивный противник, владеющий любым набором действительных групповых ключей не в состоянии вычислить никакой иной (предшествующий или последующий) групповой ключ.
5. Совершенная прямая секретность (Perfect Forward Secrecy), гарантирующая невозможность раскрытия краткосрочных групповых ключей в случае компрометации долгосрочного приватного ключа, используемого элементом группы, к примеру, для создания цифровой подписи.

Выделяют три типа методов управления групповыми ключами.

- *Централизованный*, обязывающий некий единственный объект (или набор объектов) обеспечить поддержку операций генерации и смены ключа в рамках группового взаимодействия. Методы централизованного типа не подходят в глобальных, не стойких к отказам сетевых окружениях, так как требуют постоянного присутствия и функционирования единого органа управления. Кроме того, они не подходят для групп с равноправной природой элементов. Следующие типы методов управления призваны устранить данные недостатки.

• *Распределенный*, в рамках которого элемент управления выбирается динамически при необходимости проведения операции смены ключа из списка функционирующих в данный момент элементов. Недостатком данного типа методов является необходимость установления множества двунаправленных защищенных каналов передачи данных между органом управления и каждым из элементов. При частой смене органа управления данная операция порождает существенные вычислительные потери.

• *Составной*, основанный на идее внесения эквивалентных частей общего группового ключа каждым из участников. Определенные методы данного типа (в частности, рассматриваемые в данной статье) не требуют установления защищенных каналов. Эти методы представляют наибольший интерес среди приведенных трех типов методов.

Эффективность алгоритмов рассматривается с двух точек зрения: коммуникационной и вычислительной. Коммуникационный компонент сложности формирования группового ключа учитывает быстроту выполнения протокола, т.е. количество стадий, необходимых раундов широковещания и однонаправленной передачи, а также оценивает объемы трафика, передаваемого по сети в результате выполнения протокола. Вычислительный компонент оценивает количество сложных арифметических операций (экспоненцирование, умножение, вычисление остатков и др.), необходимых для успешного выполнения протокола, как общее, так и для каждого элемента группы отдельно. Оптимальным считается протокол, достигающий минимальных показателей по данным компонентам. Однако, как показывает анализ, протоколы, эффективные по одному компоненту, как правило, имеют посредственные или негативные показатели по другому и наоборот. Кроме того, протокол может быть эффективным при поддержке одного из типов событий, к примеру, слияния и существенно проигрывать при выполнении других операций. Поэтому на практике важно использовать различные протоколы, наиболее подходящие для конкретной среды в зависимости от условий, либо же динамически изменять протокол опять-таки в зависимости от условий.

Методы формирования составных ключей на сегодня представляют собой разнообразные расширения фундаментального алгоритма, предложенного Диффи и Хеллманом в 1976 [6]. Оригинальный протокол предлагает метод установления общего ключа двумя взаимодействующими сторонами, групповые же его расширения предлагают обобщение данной схемы до случая N взаимодействующих сторон. Как и для оригинального протокола, сложность раскрытия устанавливаемого группового ключа основывается на сложности нахождения дискретного логарифма, поэтому если двусторонний протокол считается защищенным, то же самое можно утверждать для всех его расширений. Детальное описание шести наиболее известных протоколов управления групповым ключом на основе алгоритма Диффи – Хеллмана, в частности их вариантов, допускающих аутентификацию сторон, можно получить из [7-11]. Ниже приводятся основные алгоритмы и их сравнительные характеристики.

Введем следующую нотацию:

n — количество участников протокола;

i, j — индексы для участников групп;

M_i — i-й участник группы;

G – подгруппа Z_p^* порядка q, где p и q – простые;

q – порядок алгебраической группы;

a, g – образующие элементы в группе G;

x_i – долговременный секретный ключ M_i;

r_i – случайное (секретное) число $\in Z_q$, вырабатываемое M_i;

S_n – групповой ключ n участников;

$S_n(M_i)$ – вклад M_i-го участника в групповой ключ;

K_{ij} – долговременная секретная величина, выработанная M_i и M_j и $i \neq j$;

Параметры a, p, q общие для всех пользователей.

Все вычисления проводятся в циклической группе G простого порядка q, которая является подгруппой Z_p^* порядка p, где $p = kq + 1$ для некоторого $k \in N$.

Под записью α^x будем понимать операцию модулярной редукции в классах вычетов $\alpha^x \bmod p$.

Запись $M_i \to M_j$: *value* обозначает передачу участником M_i участнику M_j значений *value* в рамках протокола обмена сообщениями.

Запись $d \in_R A$ означает случайный выбор d из множества A.

Хорошо известный оригинальный алгоритм Диффи и Хеллмана имеет следующий вид.

Протокол DH

M_1, M_2 – два участника группы,

α – образующий элемент группы G, (α, p) – общий открытый ключ, известный M_1, M_2.

1. M_1 выбирает случайное число $x_1, x_1 \in_R Z_{p-1}$, вычисляет

$$y_1 = \alpha^{x_1} \bmod p,$$

$$M_1 \to M_2: y_1.$$

2. M_2 выбирает случайное число $x_2, x_2 \in_R Z_{p-1}$, вычисляет

$$y_2 = \alpha^{x_2} \bmod p,$$

$$M_2 \to M_1: y_2.$$

3. M_1 вычисляет общий ключ $K = y_2^{x_1} \bmod p = \alpha^{x_2 x_1} \bmod p$.

4. M_2 вычисляет общий ключ $K = y_1^{x_2} \bmod p = \alpha^{x_1 x_2} \bmod p$.

Существует несколько модификаций обобщения алгоритма Диффи–Хеллмана для групп. Рассмотрим некоторые из них.

В этих алгоритмах участники передают друг другу определенную информацию. В результате обменов формируется общий секретный ключ. Предполагается, что участники коалиции ранжированы и имеют известные индивидуальные идентификационные имена.

Алгоритм GDH2 является незначительно улучшенной модификацией алгоритма GDH1. Поэтому алгоритм GDH1 не рассматривается.

Алгоритм GDH2

Этап 1 – восходящая передача составляющих общего ключа.

Шаг i, $i \in [1, n\text{-}1]$.

Участник M_i передает участнику M_{i+1} набор из i+1 значений, i из которых были получены им от M_{i-1} (за исключением первого), а одно вычислено самостоятельно:

$$M_i \to M_{i+1}: \{\alpha^{\prod_{k \in [1,i], k \neq j} r_k} \mid j \in [1, i]\}, \alpha^{\prod_{k \in [1,i]} r_k}\}.$$

После завершения этапа 1 участник M_n обладает значением $\alpha^{\prod_{k \in [1,n-1]} r_k}$ и может первым вычислить общий ключ S_n.

Этап 2 – вычисление ключа абонентом M_n и передача необходимых составляющих общего ключа всем оставшимся членам группы.

Шаг n:

$$S_n = \alpha^{\prod_{k \in [1,n]} r_k}, M_n \to M_i: \{\alpha^{\prod_{k \in [1,n], k \neq i} r_k} \mid i \in$$

$$\in [1, n]\}.$$

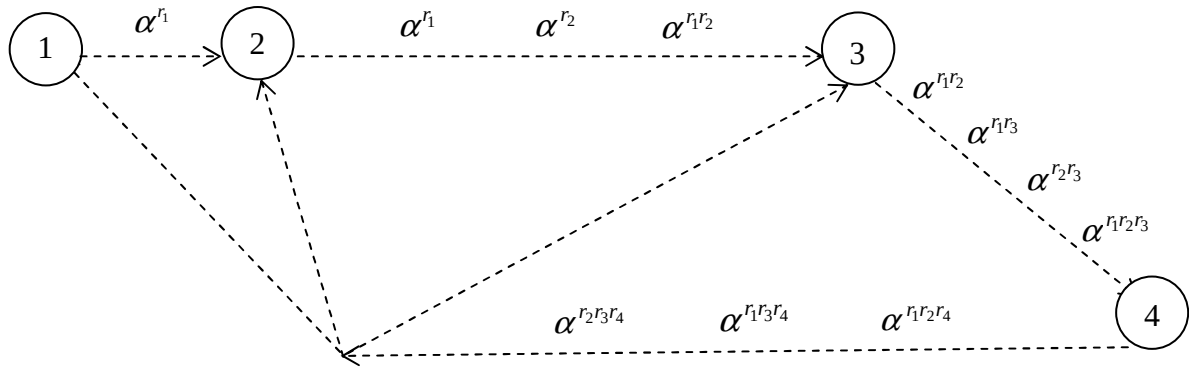

Рис 1. Схема передачи сообщений в протоколе GDH2

На рис. 1 приведен пример протокола GDH2 для четырех участников.

Алгоритм GDH3

GDH3 превосходит GDH2 по производительности благодаря меньшему количеству операций экспоненцирования, в среднем выполняемыми i-м участником группы.

Этап 1 – восходящая передача промежуточных значений.

Шаг i, $i \in [1, n\text{-}2]$:

$$M_i \to M_{i+1}: \alpha^{\prod_{k\in[1,i]} r_k}.$$

Этап 2 – широковещание.

Шаг n-1:

$$M_{n-1} \to M_i: \alpha^{\prod_{k\in[1,n-1]} r_k}.$$

Этап 3 – ответ.

Шаг n.

На данном этапе требуется удаление собственной экспоненты каждым из членов группы, что всегда возможно при простом q. Участник M_i выполняет действие $(\alpha^{r_1 \ldots r_{n-1}})^{r_i^{-1}} \bmod p$, где r_i^{-1} вычисляется из соотношения $r_i^{-1} r_i \equiv 1 \bmod q$.

$$M_i \to M_n: \alpha^{\prod_{k\in[1,n-1],k\neq i} r_k}.$$

Этап 4 – внесение своей экспоненты участником M_n, широковещание набора значений, после чего возможно вычисление общего ключа всеми участниками M_i на основании значения собственной экспоненты r_i. Знание остальных из полученного набора широковещания не является необходимым на данном этапе для отдельно взятого участника группы, но может быть необходимым для реализации протокола отделения в том случае, если в его результате данный участник окажется элементом с наибольшим индексом, среди тех, кто остаются в группе после отделения (см. протокол отделения).

Шаг n+1:

$$S_n = \alpha^{\prod_{k\in[1,n]} r_k};$$

$$M_n \to M_i: \{\alpha^{\prod_{k\in[1,n],k\neq j} r_k} \mid j \in [1, n\text{-}1]\}.$$

Данный алгоритм часто называется групповым алгоритмом Диффи–Хеллмана GDH, в силу того, что он является наиболее эффективным по вычислению алгоритмом из трех вышеперечисленных.

Ниже приводятся описания протоколов слияния и отделения для GDH3 (вступление и выход могут быть сведены к данным протоколам, поэтому не будут описываться). Как видно, слияние происходит по тому же принципу, что и установление начального ключа.

Алгоритм слияния GDH3

К группе из n участников присоединяется m новых членов группы.

Этап 1 – восходящая передача значений.

Шаг 1. M_n генерирует новую экспоненту r_n и отправляет M_{n+1} обновленное значение:

$$M_n \to M_{n+1}: (\alpha^{\prod_{k\in[1,n]} r_k}).$$

Шаг $j+1$, $j \in [1, m-1]$:

$$M_{n+j} \to M_{n+j+1}: \alpha^{\prod_{k\in[1,n+j]} r_k}.$$

Этап 2 – широковещание полученного значения участником M_{n+m} всей обновленной группе.

Шаг $m+1$:

$$M_{n+m} \to M_i: (\alpha^{\prod_{k\in[1,n+m-1]} r_k}).$$

Этап 3 – ответ.

Шаг $m+2$:

$$M_i \to M_{n+m}: \alpha^{\prod_{k\in[1,n+m-1],k\neq i} N_k}.$$

Этап 4 – широковещание и вычисление ключа всеми участниками.

Шаг $m+3$:

$$S_{n+m} = \alpha^{\prod_{k\in[1,n+m]} N_k};$$

$$M_{n+m} \to M_i: \{\alpha^{\prod_{k\in[1,n+m],k\neq j} N_k} \mid j\in [1, n+m-1]\}.$$

Алгоритм отделения GDH3

Элементы группы с множеством индексов $L \subset [1, \ldots, n]$ покидают группу.

Этап 1 – выбор координатора регенерации ключа и широковещание нового набора значений оставшимся членам группы. Вычисление всеми оставшимися участниками нового ключа. Новый координатор M_d имеет наибольший индекс среди всех оставшихся членов группы. Он вносит обновленный компонент N_d, вычисляет новый набор значений, исключая из него частичные компоненты покидающих членов группы, и распространяет его всем оставшимся участникам.

Шаг 1:

$$M_d \to M_i \mid i\notin L: \{\alpha^{\prod_{k\neq j} N_k} \mid j\notin L, j\neq d\};$$

$$S_d = \alpha^{\prod_{k\in[1..n]} N_k}.$$

Поскольку в заново распространенном наборе отсутствуют обновленные частичные ключи покинувших группу членов, они не смогут вычислить новое значение ключа.

Протокол BD (Бурместера - Десмедта)

Достоинство и главная отличительная его черта – быстрота выполнения. Теоретически он выполняется всего за 2 этапа.

Этап 1 – широковещание.

Каждый член группы M_i генерирует собственную экспоненту r и распространяет значение $z_i = \alpha^r$.

Этап 2 – широковещание и вычисление общего ключа каждым из участников группы.

Каждый участник M_i вычисляет и распространяет значение

$$x_i = (z_{i+1} / z_{i-1})^{r_i} \ (x_i = \alpha^{r_{i+1}r_i - r_i r_{i-1}}).$$

Участник M_i вычисляет

$$S_n = (z_{i-1})^{nr_i} (\prod_{j\in[0,n-2]} x_{i+j}^{n-1-j}) = \alpha^{\sum_{k,l\in[1,n],k\neq l} r_k r_l}.$$

Протокол может выполняться в 2 стадии при использовании метода одновременного широковещания более чем одним пользователем, который не всегда доступен в существующих сетевых конфигурациях.

Поэтому для построения сравнительной характеристики с протоколами следует рассматривать протокол BD* – не использующий одновременное широковещание и таким образом увеличивающий количество шагов до $2n$.

Интересной особенностью протокола является отсутствие

роли координатора и независимость от типа событий, вызвавшего смену состава. Протоколы слияния и отделения идентичны протоколу инициализации первичного ключа.

В оригинальном протоколе Диффи и Хеллмана два участника, общаясь по открытому каналу, формируют общий ключ без аутентификации друг друга.

Наиболее подходящим аутентичным протоколом формирования общего ключа, обеспечивающим свойство совершенной прямой секретности, является протокол A-DH.

Протокол A-DH

Пусть p, q, G – величины, определенные выше, и пусть α – образующий элемент G.

Предварительный этап.

Пусть x_1 и x_2 – два целых числа, $1 \le x_1, x_2 \le q-1$.

Пусть M_1 и M_2 – два участника, которые хотят выработать общий ключ и $(x_1, \alpha^{x_1} \bmod p)$ и $(x_2, \alpha^{x_2} \bmod p)$ – секретные и открытые ключи M_1 и M_2 соответственно. Открытые величины системы $(p, q, \alpha, \alpha^{x_1}, \alpha^{x_2})$. Предполагается, что α^{x_1} и α^{x_2} переданы M_1 и M_2 соответственно с помощью некоторых средств, обеспечивающих аутентификацию участников обмена.

Этап 1:

M_1 выбирает случайное $r_1 \in_R Z_q$,

$M_1 \to M_2$: $\alpha^{r_1} \bmod p$.

Этап 2:

M_2 выбирает случайное $r_2 \in_R Z_q$ и вычисляет $K = F(\alpha^{x_1 x_2} \bmod p)$,

$M_2 \to M_1$: $\alpha^{r_1 K} \bmod p$.

Когда M_1 получает $J = \alpha^{r_2 K} \bmod p$, он вычисляет $K^{-1} \bmod q$ и затем $J^{r_1 K^{-1}} \bmod p$. Получаемый в результате ключ будет $S_2 = \alpha^{r_1 r_2} \bmod p$. Функция $F(\,)$ может быть либо $F(x) = x \bmod q$, либо $F(x) = h(x)$, где h – хэш-функция: $\{0,1\}^* \to Z_q$.

Очевидно, протокол обладает контрибутивностью. В то же время обеспечивается аутентификация ключа, поскольку при его формировании участвуют открытые ключи обоих абонентов, которые, как предполагается, переданы по аутентичному каналу.

Существует несколько разновидностей аутентичного обобщения протокола Диффи–Хеллмана для групп.

Рассмотрим протокол A-GDH.2 [9].

Протокол A-GDH.2

Шаги с 1 по n-1: такой же, как и в GDH.2.

Шаг n:

M_n выбирает случайное $r_n \in_R Z_q^*$,

$M_n \to$ каждому M_i: $\{\alpha^{r_1 \ldots r_n K_{in} / r_i} \mid i \in [1, n]\}$.

При получении M_i вычисляет $\alpha^{(r_1 \ldots r_n K_{in} / r_i) K_{in}^{-1} r_i} = \alpha^{r_1 \ldots r_n} = S_n$.

В этом протоколе каждый участник группы вырабатывает общий аутентичный ключ с M_n. Более того, если мы доверяем M_n, то каждый участник группы может быть уверен, что такой же ключ имеют и все участники группы, т.е. они выработали общий групповой ключ.

Необходимо заметить, что вышеупомянутый протокол A-GDH.2 выполняет *неявную аутентификацию ключа* в довольно слабой форме, поскольку ключ не аутентифицируется непосредственно между каждыми любыми двумя участниками. Поскольку последний участник M_n отвечает за распространение аутентичных ключей K_{in}, то групповая аутентификация выполняется только при условии полного доверия M_n. Участник M_n выполняет функции доверенного сервера. Протокол A-DH нетрудно обобщить

на случай полной попарной аутентификации участников коммуникационных обменов. Как и в A-DH, предполагается, что каждая пара (M_i, M_j) имеет первоначальный ключ K_{ij}, полученный с процедурой аутентификации

Пусть R — протокол обмена для n участников и M — множество участников. Будем говорить, что R является *протоколом, обеспечивающим полную (complete) аутентификацию группового ключа*, если для каждых i, j $(0 < i \neq j \leq n)$ участники M_i и M_j вычислят общий ключ S_{ij}, *только если* S_{ij} был получен при участии каждого $M_k \subset M$.

Протокол SA-GPH.2

Шаг i $(0 < i < n)$:

1. M_i получает множество промежуточных величин $\{V_k | 1 \leq k \leq n\}$. M_1 получает пустое множество на первом этапе:

$$V_k = \begin{cases} \alpha^{(r_1 \cdots r_{i-1}/rk)(K_{k1} \cdots K_{k(i-1)})} & \text{при } k \leq i-1, k = i-1; \\ \alpha^{(r_1 \cdots r_{i-1})(K_{k1} \cdots K_{k(i-1)})} & \text{при } k > i-1. \end{cases}$$

2. M_i обновляет каждое V_k следующим образом:

$$V_k = \begin{cases} (V_k)^{r_i K_{ik}} = \alpha^{(r_1 \cdots r_i / r_k)(K_{k1} \cdots K_{ki})} & \text{при } k < i; \\ (V_k)^{r_i K_{ik}} = \alpha^{(r_1 \cdots r_i)(K_{k1} \cdots K_{ki})} & \text{при } k > i; \\ V_k & \text{при } k = i; \end{cases}$$

Шаг n:

1. M_n рассылает множество всех V_k участникам группы.

2. При получении M_i выбирает свое $V_i = \alpha^{(r_1 \cdots r_n / r_i)(K_{1i} \cdots K_{ni})}$.

M_i вычисляет $(V_k)^{r_i (K_{1i}^{-1} \cdots K_{ni}^{-1})} = \alpha^{r_1 \cdots r_n}$.

Сложностные вычислительные и коммуникационные характеристики рассмотренных алгоритмов приведены в табл. 1.

Протоколы централизованного распределения ключей CKD

Протокол CKD является типичным методом распространения групповых ключей централизованного типа.

Координатором выступает самый старший из элементов группы (имеющий наименьший индекс). Его задачей является установление двусторонних защищенных каналов с каждым из участников при согласовании начального ключа или только с новым вступающим членом группы при слиянии. По данным каналам может быть передан начальный или обновленный групповой ключ. Если в случае отделения координатор не покидает группу, установление каналов не требуется. Иначе выбирается новый координатор, который вынужден повторить процедуру установления начального ключа в рамках сокращенной группы.

Таблица 1

Стоимость вычислений	**GDH.2**	**A-GDH.2**	**SA-GDH.2**	**GDH3**	**BD***
Экспоненцирований для M_i	i+1	i+1	n	4, i < n-1 2, i = n-1	n+1
Экспоненцирований для M_n	n	n	n	n	n+1
Всего экспоненцирований	$\frac{n(n+3)}{2}-1$	$\frac{n(n+3)}{2}-1$	2n(n-1)	5n-6	n(n+1)
Коммуникационные затраты	**GDH.2**	**A-GDH.2**	**SA-GDH.2**	**GDH3**	**BD***
Количество шагов	n	n	n	n+1	2n-1
Количество сообщений	n	n	n	2n-1	2n-1

Для иллюстрации опишем протокол слияния, содержащий базовую операцию установления двухстороннего защищенного канала и передачи по нему зашифрованного значения ключа.

Алгоритм слияния CKD

К группе из n участников присоединяется m новых участников. M_1 – координатор.

Этап 1 – выбор случайной экспоненты координатором и ее широковещательная передача.

Шаг 1:

$$M_1 \to M_{n+i} \mid i \in [1..m]: \alpha^{r_1}.$$

Этап 2 – ответ

Шаг 1:

$$M_{n+i} \mid i \in [1..m] \to M_1: \alpha^{r_{n+i}}.$$

Этап 3 – выбор координатором нового ключа K_s и широковещание набора зашифрованных значений всем участникам группы. Вычисление ключа K каждым участником на основании собственной экспоненты и полученного значения α^{r_1}:

$$M_1 \to M_i \mid i \in [2..n+m]: K_s \alpha^{r_1 r_i} \bmod q.$$

В [12] построена функция с ловушкой (trap-door) для произведения двух простых чисел. Секретным параметром может быть любое произвольное число. Пусть k – произвольное натуральное число. В [12] рассмотрены две процедуры α и β. Процедура α имеет на входе число k и генерирует число $n = pq$, где p и q (большие) сильные простые числа.

Процедура β, получая на входе k и n, в качестве результата выдает числа p и q. Эта процедура не использует никаких сложных арифметических операций, в ней только дважды применяется алгоритм Евклида для нахождения наибольшего делителя двух чисел. Такая асимметричность по сложности позволяет сосредоточить функции генерации сертифицированных простых чисел в одном из участников коалиционной группы, обладающим большими вычислительными мощностями. Таким способом легко построить единую систему защиты обменов информацией типа RSA для коалиционных групп с большим составом слабых по вычислению участников.

Алгоритмы TGDH и STR

Альтернативными методами разделения общего группового криптографического ключа, отвечающими определенным групповой семантикой операциям изменения состава группы, являются методы, основанные на использовании бинарных деревьев ключей. Дерево ключей представляет собой бинарное дерево, узлы которого хранят информацию о промежуточных ключах, на основе которых вычисля-

ется общий ключ, а непосредственно участники группового взаимодействия представлены листьями дерева. В общем случае алгоритм построения бинарного дерева ключей называется алгоритмом TGDH [10]. Бинарное дерево ключей строится по следующей схеме. Для узлов дерева вводится строгая индексация. Корень находится на уровне 0 и имеет индекс (0,0). Полагаем, что узел (l,v) — v-й узел на уровне l. Потомки данного узла будут иметь индексы $(l+1, 2v)$ и $(l+1, 2v+1)$. С каждым из узлов ассоциируется пара ключей: $K_{(l,v)}$ и $BK_{(l,v)} = f(K_{(l,v)})$, являющимся скрытым ключом, где f — функция модулярного экспоненцирования в поле простых чисел, к примеру $f(k) = a^k \bmod q$ по аналогии с алгоритмом Диффи–Хеллмана. Листья дерева ассоциируются с членами группы, и для них значения $K_{(l,v)}$ выбираются случайным образом. Для промежуточных узлов значение вычисляется по следующей рекурсивной формуле:

$$K_{(l,v)} = (BK_{(l+1,2v+1)})^{K_{(l+1,2v)}} \bmod q =$$

$$= (BK_{(l+1,2v)})^{K_{(l+1,2v+1)}} \bmod q =$$

$$= a^{K_{(l+1,2v)}K_{(l+1,2v+1)}} \bmod q = f(K_{(l+1,2v)}K_{(l+1,2v+1)}).$$

Для вычисления ключа узла необходимо знание ключа одного из потомков и скрытого ключа другого. Для каждого из листов определяется путь к вершине начиная непосредственно с самого листа.

При условии владения набором скрытых ключей всех узлов, имеющих общего родителя с узлами, составляющими путь от листа к вершине, член группы, представляемый листом, имеет возможность вычисления общего группового ключа, равного значению $K_{(0,0)}$.

На рис. 2 приведен пример сбалансированного по высоте TGDH-дерева высоты 3, содержащего 6 элементов.

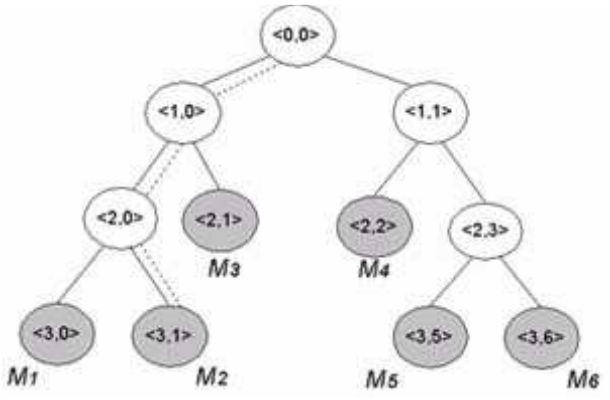

Рис. 2. Пример TGDH-дерева для 6-ти элементов.

К примеру, для участника M_2 путь составляют вершины $\{(3, 1), (2, 0), (1, 0), (0, 0)\}$. Для вычисления $K_{(0, 0)}$ необходим набор ключей $\{K_{(3, 1)}, BK_{(3, 0)}, BK_{(2, 1)}, BK_{(1, 1)}\}$.

Конечная формула вычисления группового ключа для M_2 будет иметь вид

$$M_2\colon K_{(0, 0)} = BK_{(1, 1)}^{\,BK_{(2,1)}^{\,BK_{(3,0)}^{\,K_{(3,1)} \bmod q} \bmod q} \bmod q} \bmod q.$$

Для произвольного TDGH-дерева оценка сложности операций слияния и отделения является нетривиальной задачей, поскольку количество необходимых операций экспоненцирования зависит от позиции добавления и удаления элементов.

Поддержка сбалансированности в ходе изменения состава позволяет свести вычислительные затраты генерации новых ключей в случае изменения состава к $O(\log n)$, что продемонстрировано в [10]. Однако балансировка при слиянии неравных по высоте деревьев или при удалении совокупности элементов порождает дополнительные вычислительные затраты.

Детальное обсуждение протоколов слияния и отделения для TDGH в общем случае, включая их

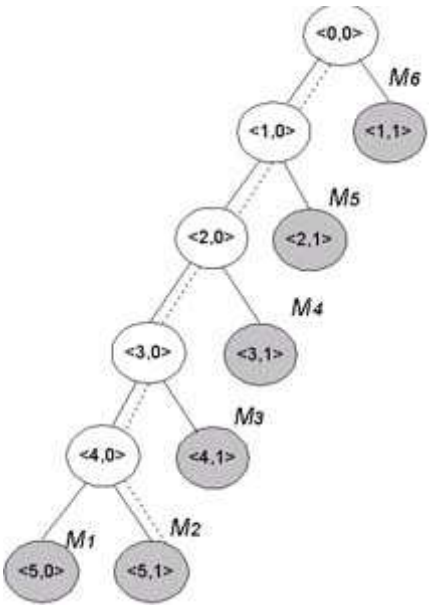

Рис. 3. Пример STR-дерева для 6-ти элементов

проблемные стороны, можно найти в [10, 13].

Существует более простая и эффективная разновидность TDGH-деревьев, STR-дерево [11], являющееся полностью несбалансированным по высоте TDGH-деревом.

Пример STR-дерева для группы из шести элементов приведен на рис. 3. Путь к вершине для члена группы M_2 указан штриховой линией.

Эффективность алгоритма STR определена однозначностью действий при операциях слияния и отделения, независимо от позиций вовлеченных элементов, а также малым количеством сообщений, необходимых для регенерации ключей.

Для дерева, приведенного на рис. 3 конечная формула для M_2 принимает вид

$$M_2: K_{(0,0)} = BK_{(1,1)}^{\,BK_{(2,1)}^{\,BK_{(3,1)}^{\,BK_{(4,1)}^{\,BK_{(5,0)}^{\,K_{(5,1)}}}}}} \bmod q$$

$$\bmod q \bmod q \bmod q \bmod q \bmod q.$$

Протокол слияния STR

Допустим, дерево, содержащее элементы $M_i | \, i \in [1,...,4]$, сливается с деревом, содержащим $M_j | \, j \in [5,...,7]$ (рис. 4).

Правые потомки корней каждого из деревьев M_4 и M_7 выступают в роли координаторов слияния. Они вычисляют скрытый групповой ключ своего дерева и широковещательно передают полный набор скрытых ключей своего дерева другой стороне. После его получения каждый элемент обоих деревьев перестраивает дерево как следует из рисунка (дерево меньшей высоты считается присоединяемым к дереву большей высоты).

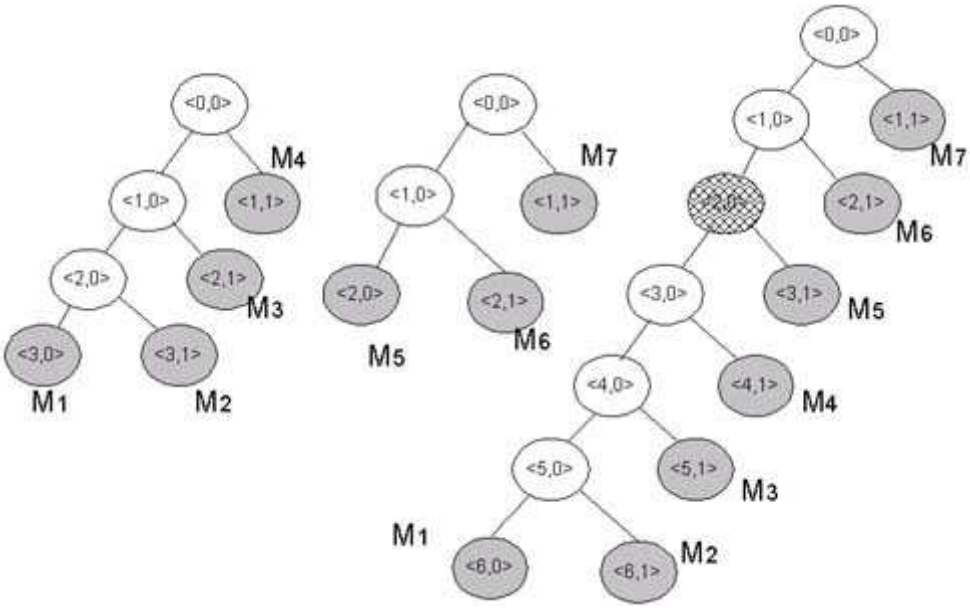

Рис. 4. Схема протокола слияния STR

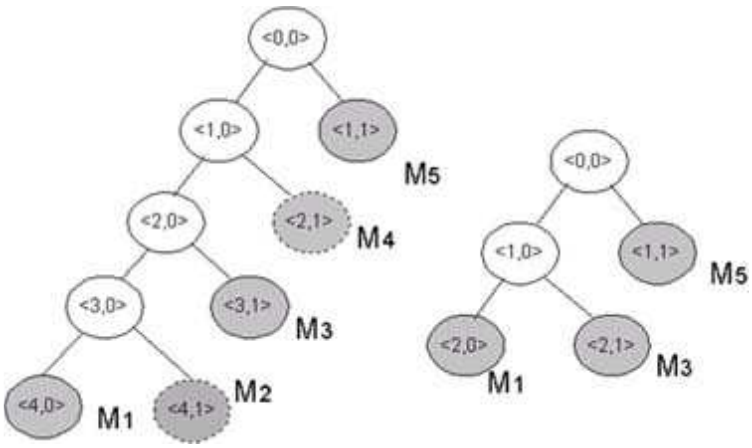

Рис. 5. Схема протокола отделения STR

Появляется новый узел на позиции <2,0>, левым потомком которого становится бывший корень дерева большей высоты, а правым — элемент с минимальным индексом из меньшего дерева M_5, который становится новым координатором. После этого новый координатор вычисляет изменившиеся в результате слияния скрытые ключи и широковещательно передает их всем элементам, которые получают возможность вычислить обновленный групповой ключ.

Протокол отделения STR

Совокупность участников с множеством индексов $L \subset [1, ..., n]$ покидает группу. Координатором выступает элемент, находящийся непосредственно под покидающим элементом с наименьшим индексом. Если M_1 также покидает группу, то координатором будет элемент с наименьшим индексом среди оставшихся.

Оставшиеся элементы перестраивают дерево, как показано на рис. 5, координатор изменят собственную экспоненту, вычисляет изменившиеся скрытые ключи и распространяет их значения. Получив данный набор, все элементы могут вычислить новый групповой ключ.

Выводы

Данная статья является началом обзора по тематике динамических коалиций — нового поколения защищенных систем партнерского взаимодействия. Была раскрыта актуальность тематики, описаны основные составляющие коалиционного окружения и приведены основные алгоритмы групповой криптографии. Данные алгоритмы используются для обеспечения информационной защиты в процессе группового сотрудничества и отвечают условиям динамичности коалиционной среды. В

следующей части обзора будут рассмотрены практические подходы к реализации динамических коалиционных сред на базе существующих распределенных приложений.

1. *Scaling* Secure Group Communication: Beyond Peer-to-Peer. / Y. Amir, C. Nita-Rotaru, J. Stanton, G. Tsudik. Proc. of DISCEX III Conf. — Washington DC, April 2003. — P. 12.
2. *Таненбаум Э., Ван Стеен*. Распределенные системы: принципы и парадигмы. Изд.дом. "Питер", 2003. — С.
3. *Dynamic* PKI and Secure Tuplespaces for Distributed Coalitions. / T.J. Smith, G.T. Byrd, X. Wu, H. Xin, K.Thangavelu, R. Wang, A.Shah. Proc. of DISCEX III Conf. — Washington DC, April 2003. — P. 12.
4. *Integrated* Security Services for Dynamic Coalitions / H. Khurana, S. Gavrila, R. Bobba, R. Koleva, A. Sonalker, E.Dinu, V. Gligor, J. Baras. Proc. of DISCEX III Conf. — Washington DC, April 2003. — P. 3.
5. *Efficient* and Secure Information Sharing in Distributed, Collaborative Environments / P. Dasgupta, V. Karamcheti, Z. Kedem. Proc. of the Third Intern. Workshop on Communication-based Systems, Berlin, March 2000. — P. 16
6. *Diffie W., Hellman M.* New directions in cryptography // IEEE Trans. on Information Theory. — 1976. — IT-22. — N6. — P. 644–654.
7. *Steiner M., Tsudik G., Waidner M.* Diffie-helman key distribution extended to groups // Proc. of the 3rd ACM Conf. on Computer and Communication Security. — 1996. — March. — P. 31–37.
8. *Burmester M., Desmedt Y.* A secure and efficient conference key distribution system // Advances in Cryptology. — EUROCRYPT’94. — Lecture Notes in Computer Science. — Berlin etc.: Springer-Verlag, 1994. — Vol. 950. — P. 267–275.
9. *Ateniese G., Steiner M., Tsudik G.* Authenticated Group Key Agreement and Friends // Proc. of the 5-th ACM Conf. on Computer and Communication Security, San Francisco, November 1998. — P. 17–26.
10. *Kim Y., Perrig A., Tsudik G.* Simple and Fault-Tolerant Key Agreement for Dynamic Collaborative Groups // Proc. of the 7-th Annual ACM Conf. on Computer and Communication Security, November 2000. — P. 235–244.
11. *Kim Y., Perrig A., Tsudik G.* Communication-Efficient Group Key Agreement // Proc. of Intern. Federation for Information Processing (IFIP) SEC, June 2001. — P. 229–244.
12. *Анисимов А.В.* Коалиционные схемы с ключами общего доступа // Кибернетика и системный анализ. — 2001. — № 1. — С. 3–18.
13. *On the Performance of* Group Key Agreement Protocols // Y. Amir, Y. Kim, C. Nita-Rotaru, G. Tsudik. Proc. of ICDCS’2002. — 2002. — July P. 23.
14. *Dynamic* Cryptographic Context Management (DCCM). — Report 3. Cryptographic Context Negotiation Protocol / D.M. Balenson, D.K. Branstad, P.T. Dinsmore, M. Heyman, C. Scase. TISR #0709. — TIS Labs at Network Associates, Inc. — 1999, February. — 12 p.
15. *Stallings W.* Cryptography and Network Security: Principles and Practice. — Prentice Hall. — 1999.

Получено ***06.05.04***

Об авторах

Анисимов Анатолий Васильевич
д-р физ.-мат. наук, профессор, декан ф-та кибернетики

Зубенко Антон Витальевич
аспирант

Место работы авторов:
Киевский национальный университет им. Т. Шевченко, Киев, Украина.

Тел. (044) 259 0427
E-mail: ava@mi.unicyb.kiev.ua
zubenko@softhome.net