Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей (05.13.11)

УДК 004.057.4
DOI: 10.24160/1993-6982-2019-1-108-113

Сетевые протоколы распределенного вычисления ключей с использованием протокола Керберос

А.М. Винников, А.Б. Фролов

Изучены криптографические протоколы обмена ключевой информацией в компьютерной сети, содержащей доверенный центр (сервер) и ряд пользователей, разделяющих с сервером индивидуальные ключи симметричной системы шифрования. Имеются множества привилегированных групп и отчужденных коалиций пользователей. На сервере по определенной схеме предварительного распределения ключей вычисляются предварительные ключи, по которым пользователи любой привилегированной группы могут определить общий рабочий ключ, недоступный пользователям любой отчужденной коалиции, не пересекающейся с данной привилегированной группой пользователей. Предложена общая структура протоколов, в которых с использованием обновляемой системной ключевой информации вычисляются предварительные ключи, доставляемые пользователям и используемые ими для получения рабочих ключей. Изучаемые протоколы наследуют структуру и функциональность протокола с метками времени, распространяя их на этап доставки пользователям предварительных ключей и данных, позволяющих находить общий рабочий ключ для коммуникаций внутри группы. Данные протоколы обеспечивают взаимную аутентификацию пользователей каждой привилегированной группы в определенный момент времени и устойчивость к атакам участников отчужденных групп, то есть невозможность выступить в роли члена привилегированной группы или навязывать устаревший или скомпрометированный ключ.

Ключевые слова: криптографический протокол, схема предварительного распределения ключей, привилегированная группа, отчужденная коалиция, схема Блома, KDP-схема, Керберос.

Для цитирования: Винников А.М., Фролов А.Б. Сетевые протоколы распределенного вычисления ключей с использованием протокола Керберос // Вестник МЭИ. 2019. № 1. С. 108—113. DOI: 10.24160/1993-6982-2019-1-108-113.

Network Protocols for Distributed Computing of Keys using the Kerberos Protocol

A.M. Vinnikov, A.B. Frolov

Cryptographic protocols for exchange of key information in a computer network containing a trusted center (server) and a number of users sharing individual keys of a symmetric encryption system with the server are studied. There are sets of privileged user groups and sets of alienated user coalitions. Preliminary keys are computed on the server according to a certain preliminary key distribution scheme, from which the users of any privileged group can determine a common working key that is not accessible to the users of any alienated coalition that does not overlap with this privileged user group. A general structure of protocols is proposed, in which the preliminary keys delivered to the users and used by them to obtain the working keys are computed using the updated system key information. The studied protocols inherit the structure and functionality of the protocol with time stamps and extend them for the stage of delivering to users the preliminary keys and data from which the users can compute a common working key for communications within the group. Such protocols ensure mutual authentication of users in each privileged group at a certain moment of time and resistance to attacks from the members of alienated groups, that is, impossibility to behave as a privileged group member or to impose an outdated or compromised key.

Key words: cryptographic protocol, key pre-distribution scheme, privileged group, alienated coalition, Blom scheme, key distribution pattern, Kerberos.

For citation: Vinnikov A.M., Frolov A.B. Network Protocols for Distributed Computing of Keys using the Kerberos Protocol. MPEI Vestnik. 2019;1:108—113. (in Russian). DOI: 10.24160/1993-6982-2019-1-108-113.

Введение

Управление ключами — одна из основных проблем обеспечения безопасности коммуникаций в компьютерных сетях. Она включает в себя задачи генерации и распределения ключей, их обновление и замену скомпрометированных ключей. Существует широкий спектр протоколов для распределения ключей, сгенерированных сервером в их окончательной форме, с использованием симметричных секретных каналов [1, 2]. Наиболее популярен протокол Needham-Schroeder [2], модификация с использованием временных меток которого лежит в основе протокола Kerberos [3].

Для уменьшения количества секретной ключевой информации, передаваемой по каналам, процесс вычисления ключей распределяется между сервером и пользователями [4, 5]. В настоящее время актуален вопрос распределения ключей в беспроводных сетях датчиков, объединяющий множество датчиков и исполнительных органов в промышленных системах, безопасность которых поддерживается криптографическими средствами [6 — 8]. Только они могут обеспечить, например, конфиденциальный учет интеллектуальных измерений с несколькими потребителями данных [9 — 11].

Изучаемая в настоящей работе компьютерная сеть состоит из доверенного сервера T и множества $\mathbf{U} = \{1, 2, ..., n\}$ пользователей, зарегистрированных на сервере и использующих совместно с ним свои индивидуальные ключи $K_{T,i}$, $i = 1, \ldots, n$, симметричной криптосистемы. Обозначим через $\mathbf{P}$ множество привилегированных групп P пользователей, а через $\mathbf{F}$ — набор отчужденных коалиций F пользователей. На сервере T вычисляются предварительные ключи (далее — ключи) $\mathbf{k}_i$ пользователей $i \in P$, согласно которым пользователи из P могут вычислять общий рабочий ключ k_P, недоступный пользователям любой коалиции F, не пересекающейся с P. Ключи $\mathbf{k}_i$ определяются на сервере по некоторой схеме предварительного распределения ключей, обозначенной в общем случае ($\mathbf{P}$,$\mathbf{F}$)-KPS (Key Predistribution Scheme). Подобными схемами являются, например, схема Блома [4, 12], KDP-схема (Key Distribution Pattern) [4] и их модификации [13]. В ряде работ описаны схемы, безопасность которых зависит от сложности нахождения прообраза значения хэш-функции [14], так, в [15] эти подходы реализуются в определенном сочетании.

Во многих работах основное внимание уделяется оценкам информационной скорости схем предварительного распределения ключей и их безопасности, считается, что рассчитанные на сервере по конкретной ($\mathbf{P}$, $\mathbf{F}$)-KPS (или по (2,1)-KPS — частному случаю, когда для всех $P \in \mathbf{P}$ и $F \in \mathbf{F}$ выполняется $|P|=2$, $|F|=1$) ключи $\mathbf{k}_i$ доставляются пользователям «offband», т. е. по той или иной секретной методике без использования сети.

В настоящей работе рассмотрены криптографические протоколы, по которым ключи доставляются пользователям посредством соответствующих транзакций в единой компьютерной сети. Проанализированы протоколы с временными отметками, обеспечивающие взаимную аутентификацию пользователей каждой группы P в определенный момент времени T и безопасность атак членов коалиции F, т. е. неспособность действовать как член группы P или навязать скомпрометированный ключ. Протоколы наследуют структуру и функциональность базового протокола с аутентификацией сообщений с использованием временных меток [2], распространяя их на этап доставки пользователям их индивидуальных ключей $\mathbf{k}_i$ и сопутствующих данных, позволяющих вычислять рабочие ключи для обмена сообщениями в группах P.

Цель работы — обоснование общей структуры протоколов, в которых с использованием обновляемой системной ключевой информации вычисляются ключи $\mathbf{k}_i$, доставляемые пользователям и используемые ими для получения рабочих ключей. Составляющими структуры являются известные криптографические примитивы, отвечающие за генерацию случайных последовательностей, реализацию схем вычисления предварительных ключей того или иного рода, сервисы протокола Kerberos формирования и распределения долгосрочных ключей и формирования транзакций с ключами $\mathbf{k}_i$, протоколы доставки этих транзакций пользователям с контролем целостности. Предлагаемое решение инвариантно относительно конкретных реализаций примитивов, в частности, версии протокола Kerberos. В отличие от [16] взят протокол Kerberos в неизменном (немодифицированном) виде с добавлением сервиса, который использует протокол Kerberos через Generic Security Services Application Program Interface GSSAPI [17].

Вычисление обновляемой системы предварительных ключей

Ключи $\mathbf{k}_i$ вычисляются на сервере на основе некоторой секретной информации $\mathbf{K}$, обновляемой для каждого сеанса в формате определенной ($\mathbf{P}$, $\mathbf{F}$)-KPS. В этом случае сервер использует систему $\boldsymbol{ID}$ идентификаторов ключей, характерную для данной ($\mathbf{P}$, $\mathbf{F}$)-KPS. Последняя позволяет на основе секретной информации $\mathbf{K}$ по общедоступному правилу $\mathrm{R}_{(\mathbf{P},\,\mathbf{F})\text{–KPS}}$, характерному для данной ($\mathbf{P}$, $\mathbf{F}$)-KPS, рассчитать соответствующие идентификаторам секретные индивидуальные ключи $\mathbf{k}(id, \mathbf{K}) = \mathrm{R}_{(\mathbf{P},\,\mathbf{F})\text{–KPS}}(id, \mathbf{K})$. Система $\boldsymbol{ID}$ и это правило составляют формальное описание конкретной ($\mathbf{P}$, $\mathbf{F}$)-KPS.

Назовем описанную инструкцию для вычисления индивидуальных ключей по секретной ключевой информации $\mathbf{K}$ *алгоритмом* данной ($\mathbf{P}$, $\mathbf{F}$)-KPS.

Индивидуальные ключи назначаются конкретным пользователям по инъективной таблице $\mathbf{I}:\mathbf{U}\rightarrow\boldsymbol{ID}$, $|\boldsymbol{ID}| \geq n$. Тогда индивидуальные ключи $\mathbf{k}_i = \mathrm{R}(\mathbf{I}(i), \mathbf{K})$ становятся ключами $\mathbf{k}_i$ пользователей i, $i = 1, \ldots, n$. Таблица $\mathbf{I}$ и, следовательно, пары $(i, \mathbf{I}(i))$ общедоступны, но должны быть защищены от несанкционированной модификации.

Каждый пользователь из группы P по своему ключу $\mathbf{k}_i$ и набору пар $\{(i, \mathbf{I}(i)):i\in P\}$ может вычислить общий ключ k_P этой группы в соответствии с общедоступным правилом $k_P=\mathrm{R}'_{(\mathbf{P},\mathbf{F})-\mathrm{KPS}}\left(\mathbf{k}_i,\left\{\left(i,\mathbf{I}\left(i\right)\right):i\in P\right\}\right)$. Таким образом, для вычисления рабочего секретного ключа для группы P достаточно знать ключ $\mathbf{k}_i$ любого из ее членов и защищенную от неавторизованной модификации таблицу $\mathbf{I}$, а также состав группы или соответствующую часть ***IP*** таблицы $\mathbf{I}$.

В общем случае система ключей, а затем рабочие ключи на основе обновляемой системной ключевой информации могут быть рассчитаны в соответствии с общей схемой с учетом особенностей данной схемы предварительного распределения ключей.

Один протокол доставки предварительных ключей, полученных по (2, 1)-KPS

В таблице представлен сетевой протокол распределения ключей на основе (2, 1)-KPS с отметками времени T.

По этому протоколу сервер T по запросу инициатора $A\in\mathbf{U}$ сеанса вычисляет на основе обновленной системной ключевой информации $\mathbf{K}$ по правилам определенной (2, 1)-KPS ключи, доставляемые пользователям. Два пользователя i и j, $j\in P(i)$ на основе ключей $\mathbf{k}_i$ или $\mathbf{k}_j$, полученных от сервера вместе с общедоступным, но защищенным от модификации описанием таблицы $\mathbf{I}_i$, находят общий ключ $k_{\{i,j\}}$, недоступный для любого другого пользователя.

Данным протоколом подтверждается возможность реализации предлагаемой общей структуры генерации, доставки и использования предварительных ключей при наличии сервера, обеспечивающего систему этими и долговременными ключами для их доставки.

Формирование и доставка предварительных ключей на основе протокола Kerberos

Для реализации анализируемого подхода к распределению ключей в компьютерной сети необходимо обеспечить регистрацию пользователей и обеспечить доставку им долговременных ключей $K_{T,i}$, периодическое обновление системной ключевой информации $\mathbf{K}$, вычисление на ее основе ключей $\mathbf{k}_i$ и доставку их пользователям.

Опишем данный процесс на базе протокола Kerberos 5 — протокола обеспечения безопасной аутентификации пользователей в незащищенной среде, где передаваемые пакеты могут быть перехвачены и модифицированы.

Рассмотрим сетевой обмен между тремя сторонами: пользователем i (клиентом), доверенным центром аутентификации, в терминах протокола Kerberos — Key Distribution Center (KDC, центр распределения ключей) и сервером T, на котором работает служба формирования предварительных ключей, и доступ к

Протокол

Этап	Данные и передачи
1	$A{:}K_{T,A}$; $A\to T{:}[A]_{K_{T,A}}$
2	T: (**2**, m)-KPS, $\mathbf{I}$, A,T, $\mathbf{K}\in_R F_q^{\tau}$
2.1	for all $i\in\mathbf{U}{:}\mathbf{k}_i\leftarrow(i, \mathbf{I}_i, \mathrm{T}, \mathbf{K})$
2.2	$T\to A$: for all $i\in\mathbf{U}{:}[i, \mathbf{I}_i, \mathrm{T}, \{k_i\}_{K_{T,i}}]_{K_{T,i}}$
3	A: $K_{T,A}$, for all $i\in\mathbf{U}{:}[i, \mathbf{I}_i, \mathrm{T}, \{k_i\}_{K_{T,i}}]_{K_{T,i}}$, $\mathbf{I}_A=\mathbf{I}$, T, $\mathbf{k}_A\leftarrow\{k_A\}_{K_{T,A}}$, if $\mathbf{t}<\mathrm{T}+\Delta_1+\Delta_2$: for all $i\in\mathbf{U}, i\neq A$: $A\to i: A,[i,\mathbf{I}_i,\mathrm{T},\{k_i\}_{K_{T,i}}]_{K_{T,i}}$;
4	for all $i\neq A$: i, $K_{T,i}$, i^R, $\mathbf{I}_i$, T, $\mathbf{k}_i\leftarrow\{\mathbf{k}_i\}_{K_{T,i}}$,
4.1	if $i=i^R$ and $t<\mathrm{T}+\Delta_1+\Delta_2$:
4.2	$N_i\in_R F_q$, for $j\subseteq P(i)$: $\mathbf{k}_{\{i,j\}}\leftarrow(j, \mathbf{I}_j, \mathbf{k}_i)$; $i\to j{:}\{j, i\}, [i, N_i]_{\mathbf{k}_{\{i,j\}}}$; $j{:}\mathbf{I}, \mathbf{k}_j, i, \mathbf{k}_{\{i,j\}}\leftarrow(i, \mathbf{I}_i, \mathbf{k}_j)$; i^R, N_i^R; if $i=i^R$: $j{:}r=N_i^R-1; j\to i{:}[r]_{\mathbf{k}_{\{i,j\}}}$;
4.3	$i{:}r^R$, $r^R=N_i-1$?

Примечание: $[\langle\text{message}\rangle]_k$ — сообщение, защищенное от модификации ключом k; $\{\langle\text{message}\rangle\}_k$ — сообщение, зашифрованное на ключе k; $A\to B$: <message> — означает передачу сообщения <message> от A к B; U: <data> — значит, что U имеет данные <data>; U: <data 1>←<data 2> — означает, что участник U вычислил данные <data 1>, используя <data 2>; $\boldsymbol{t}$ — локальное время получателя; Δt_1 — расхождение между временем сервера и локальным временем; Δt_2 — ожидаемое время сетевой задержки; <data>^{R} — копия <data>; $P(i)\subseteq U$ — множество участников, доступных участнику i; $\mathbf{K}\in_R F_q^{\tau}$ — случайный выбор τ элементов поля F_q

которому требуется клиенту. KDC ведет базу учетных данных с информацией обо всех пользователях и службах (principal), для каждого принципала сохраняется криптографический ключ, известный только этому объекту и службе KDC. Указанный долговременный ключ используется для связи пользователя системы с центром распределения ключей.

В большинстве практических реализаций протокола Kerberos долговременные ключи шифрования формируются на основе пароля пользователя в результате хэширования. В процессе первичного обмена пользователь передает KDC зашифрованное своим ключом время и имя принципала, KDC расшифровывает и проверяет сообщение, затем отправляет пользователю ключ сессии, посредством которого будет обеспечи-

ваться шифрование данных при любом дальнейшем обмене между клиентом и службой KDC, а также билет на получение билета Ticket-GrantingTicket (TGT).

Первоначальный обмен обычно осуществляется пользователем командой «kinit<имя принципала>» из дистрибутива MITKerberos 5. Команда запрашивает пароль и сохраняет выданный TGT на диск для использования другими командами, в частности библиотекой GSSAPI в программах на python. Для осуществления первоначального обмена в C++ или python без вызова консольных утилит Kerberos можно использовать низкоуровневый интерфейс библиотеки krb5 [3] или PyKerberos [18]. Клиент обращается к службе KDC за доступом к ресурсам на сервере *T*, используя свой TGT и маркер времени, которые зашифрованы с помощью ключа сессии. KDC создает пару билетов, один для клиента, один для сервера *S*. Каждый билет содержит имя принципала, запрашивающего доступ, имя получателя запроса *T*, маркер времени, а также срок жизни билета. Оба билета также содержат новый симметричный ключ $K_{T,i}$, известный и клиенту, и серверу. Данный ключ обеспечивает безопасность взаимодействия между ними. KDC шифрует билет сервера, используя долговременный ключ сервера, затем вкладывает билет сервера внутрь билета клиента, который также содержит ключ $K_{T,i}$,. Получив билет, клиент зашифровывает маркер времени с помощью ключа $K_{T,i}$, затем отправляет его и билет сервера самому серверу *T*. Теперь и клиент, и сервер обладают ключом $K_{T,i}$. Следовательно, сервер может быть уверен в том, что клиент правильно идентифицирован, поскольку для шифрования маркера времени был использован $K_{T,i}$,. В случае необходимости ответа сервера клиенту, сервер воспользуется ключом $K_{T,i}$, и клиент будет знать, что сервер правильно идентифицирован.

Взаимодействие сервера с клиентом осуществляется через UDP-сокет и GSSAPI. После согласования ключа между сервером и клиентом сервер *T* формирует для всех $i \in \mathbf{U}$ сообщения $M = [i, \mathbf{I}_i, T, \{k_i\}_{K_{T,i}}]_{K_{T,j}}$, используя полученные по алгоритму (**P**, **F**)-KPS индивидуальные ключи и таблицу **I** для вычисления ключей $\mathbf{k}_i$. Защита от модификации обеспечивается сопровождением сообщения *M* значением ключевой хэш-функции hmac($K_{T,i}$, *M*) (используется HMAC-MD5). Конечно, такое сообщение имеет смысл отправлять клиенту только когда у сервера *T* будут согласованы все ключи $K_{T,i}$ со всеми пользователями. Клиент, завершив идентификацию, ожидает сообщения от сервера. Затем он выбирает по номеру идентификатор своего ключа, для расшифровки $\{k_i\}_{K_{T,i}}$ использует ключ $K_{T,i}$ и вычисляет ключи $K_{i,j}$. Таким образом, сервисы Kerberos и сервер *T* достаточны для реализации обоснованной в работе общей структуры генерации, доставки и использования предварительных ключей. Сервер приложений протокола Kerberos необходим для вычисления и распределения предварительных ключей [19]. В предложенном варианте он формирует и, с использованием долговременных ключей симметричной системы, создаваемых KDC Kerberos, пересылает участникам сети в защищенном виде пакеты данных, достаточные для вычисления рабочих ключей.

Заключение

Обоснована общая структура сетевых протоколов распределения ключей с использованием схем их предварительного распределения, инвариантная к версиям используемых криптографических примитивов и реализуемая благодаря широко употребляемым программным средствам. Дальнейшее изучение протоколов распределенного вычисления ключей в компьютерной сети предполагается осуществить с учетом особенностей сенсорных сетей [20].

Литература

1. **Stinson D.R.** Cryptography. Theory and Practice. Boca Raton: CRC Press, 2006.

2. **Wenbo Mao.** Modern Cryptography. Theory and Practice. New Jersey: Hewlett-Packard Company. Prentice Hall, Inc. Upper Saddle River, (2003).

3. **MIT Kerberos** Documentation. Complete Reference. API and Datatypes [Электрон. ресурс] https://web.mit.edu/kerberos/krb5-1.14/doc/appdev/refs/index.html (дата обращения 05.03.2018).

4. **Алферов А.П., Зубов А Ю., Кузьмин А.С., Черемушкин А.В.** Основы криптографии. М.: Гелиос АРВ, 2005.

5. **Лось А.Б., Нестеренко А.Ю. Рожков М.И.** Криптографические методы защиты информации. М.: Юрайт, 2017.

6. **Akhbarifar S., Rahmani A.M.** A Survey on Key Pre-distribution Schemes for Security in Wireless.

References

1. **Stinson D.R.** Cryptography. Theory and Practice. Boca Raton: CRC Press, 2006.

2. **Wenbo Mao.** Modern Cryptography. Theory and Practice. New Jersey: Hewlett-Packard Company. Prentice Hall, Inc. Upper Saddle River, (2003).

3. **MIT Kerberos** Documentation. Complete Reference. API and Datatypes [Elektron. Resurs] https://web.mit.edu/kerberos/krb5-1.14/doc/appdev/refs/index.html (Data Obrashcheniya 05.03.2018).

4. **Alferov A.P., Zubov A Yu., Kuz'min A.S., Cheremushkin A.V.** Osnovy Kriptografii. M.: Gelios ARV, 2005. (in Russian).

5. **Los' A.B., Nesterenko A.Yu. Rozhkov M.I.** Kriptograficheskie Metody Zashchity Informatsii. M.: Yurayt, 2017. (in Russian).

6. **Akhbarifar S., Rahmani A.M.** A Survey on Key Pre-distribution Schemes for Security in Wireless.

Sensor Networks // Intern. J. Computer Networks and Communications Security. 2014. V. 2. No. 12. Pp. 423—442.

7. **Xiaomin Li e. a.** A Review of Industrial Wireless Networks in the Context of Industry 4.0 // Wireless Network. 2017. V. 23. Pp. 23—41.

8. **Di-Pietro R., Guarino S., Verde N.V., Domingo-Ferrer J.** Security in Wireless Ad-hoc Networks: a Survey // Computer Communications. 2014. V. 51. Pp. 1—20.

9. **Rottondi C., Verticale G., Capone A.** Privacy-preserving Smart Metering with Multiple Data Consumers // Computer Networks. 2013. V. 57. Pp. 1699—1713.

10. **Ma C., Xue K., Hong P.** Distributed Access Control with Adaptive Privacy Preserving Property for Wireless Sensor Networks // Security and Communication Networks. 2014. V. 7 (4). Pp. 759—773.

11. **Oualha N., Olivereau A.** Sensor and Data Privacy in Industrial Wireless Sensor Networks // Proc. Conf. Network and Information Systems Security. La Rochelle. 2011. Pp. 1—8.

12. **Stinson D.R.** Overview of Attack Models and Adversarial Goals for SKDS and KAS. CS 758 [Электрон. ресурс] http://cgi.di.uoa.gr/~halatsis/Crypto/Bibliografia/Crypto_Lectures/Stinson_lectures/lec09.pdf (дата обращения 05.03.2018).

13. **Shruthi P., Nirmala M.B., Manjunath A.S.** Secured Modified Bloom's based Q-composite Key Distribution for Wireless Sensor Networks // Intern. J. Advanced Computer Theory and Eng. 2013. V. 2. No. 5. Pp. 2319—2526.

14. **Ramkumar M.** Symmetric Cryptography Protocols. Heidelberg, N.-Y., Dordrecht, London: Springer, 2014.

15. **Frolov A., Zatey A.** Probabilistic Synthesis of KDP Satis-fying Mutually Complementary Correctness Conditions // Proc. Intern. Conf. Advances in Computing, Communication and Information Techn. 2014. Pp. 75—79.

16. **Frolov A.B., Shchurov I.I.** Non-centralized Key Pre-distribution in Computer Networks // IEEE Proc. Intern. Conf. Dependability of Computer Syst. 2008. Pp. 179—188.

17. **Generic** Security Services Application Program Interface — Python GSSAPI wrapper [Офиц. сайт] https://pypi.python.org/pypi/gssapi (дата обращения 05.03.2018).

18. **The Python** Package Fora High-level Wrapper for Kerberos (GSSAPI) Operations [Офиц. сайт] https://pypi.python.org/pypi/pykerberos/1.1.14 (дата обращения 05.03.2018).

19. **Щуров И.И.** Методы и программные средства предварительного распределения ключей в компьютерной сети: дисс. … канд. техн. наук. М.: Изд-во МЭИ, 2008.

20. **Chi-Yua Chen, Han-Chieh Chao.** A Survey of Key Distribution in Wireless Sensor Networks // Security and Communication Networks. 2014. V. 7. No. 12. Pp. 2495—2508.

Sensor Networks. Intern. J. Computer Networks and Communications Security. 2014;2;12:423—442.

7. **Xiaomin Li e. a.** A Review of Industrial Wireless Networks in the Context of Industry 4.0. Wireless Network. 2017;23:23—41.

8. **Di-Pietro R., Guarino S., Verde N.V., Domingo-Ferrer J.** Security in Wireless Ad-hoc Networks: a Survey. Computer Communications. 2014;51:1—20.

9. **Rottondi C., Verticale G., Capone A.** Privacy-preserving Smart Metering with Multiple Data Consumers. Computer Networks. 2013;57:1699—1713.

10. **Ma C., Xue K., Hong P.** Distributed Access Control with Adaptive Privacy Preserving Property for Wireless Sensor Networks. Security and Communication Networks. 2014;7 (4):759—773.

11. **Oualha N., Olivereau A.** Sensor and Data Privacy in Industrial Wireless Sensor Networks. Proc. Conf. Network and Information Systems Security. La Rochelle. 2011:1—8.

12. **Stinson D.R.** Overview of Attack Models and Adversarial Goals for SKDS and KAS. CS 758 [Elektron. Resurs] http://cgi.di.uoa.gr/~halatsis/Crypto/Bibliografia/Crypto_Lectures/Stinson_lectures/lec09.pdf (Data Obrashcheniya 05.03.2018).

13. **Shruthi P., Nirmala M.B., Manjunath A.S.** Secured Modified Bloom's based Q-composite Key Distribution for Wireless Sensor Networks. Intern. J. Advanced Computer Theory and Eng. 2013;2;5:2319—2526.

14. **Ramkumar M.** Symmetric Cryptography Protocols. Heidelberg, N.-Y., Dordrecht, London: Springer, 2014.

15. **Frolov A., Zatey A.** Probabilistic Synthesis of KDP Satis-fying Mutually Complementary Correctness Conditions. Proc. Intern. Conf. Advances in Computing, Communication and Information Techn. 2014:75—79.

16. **Frolov A.B., Shchurov I.I.** Non-centralized Key Pre-distribution in Computer Networks. IEEE Proc. Intern. Conf. Dependability of Computer Syst. 2008:179—188.

17. **Generic** Security Services Application Program Interface — Python GSSAPI wrapper [Ofits. Sayt] https://pypi.python.org/pypi/gssapi (Data Obrashcheniya 05.03.2018).

18. **The Python** Package Fora High-level Wrapper for Kerberos (GSSAPI) Operations [Ofits. Sayt] https://pypi.python.org/pypi/pykerberos/1.1.14 (Data Obrashcheniya 05.03.2018).

19. **Shchurov I.I.** Metody i Programmnye Sredstva Predvaritel'nogo Raspredeleniya Klyuchey v Komp'yuternoy Seti: Diss. … Kand. Tekhn. Nauk. M.: Izd-vo MEI, 2008. (in Russian).

20. **Chi-Yua Chen, Han-Chieh Chao.** A Survey of Key Distribution in Wireless Sensor Networks. Security and Communication Networks. 2014;7;12:2495—2508.

Сведения об авторах:

Винников Александр Михайлович — заместитель начальника отдела материально-технического обеспечения ООО «СофтИнжиниринг», e-mail: al.vin@bk.ru
Фролов Александр Борисович — доктор технических наук, профессор кафедры математического моделирования НИУ «МЭИ», e-mail: abfrolov@mail.ru

Information about authors:

Vinnikov Aleksandr M. — Deputy Head of Logistics Dept., OOO «Soft Engineering», e-mail: al.vin@bk.ru
Frolov Aleksandr B. — Dr.Sci. (Techn.), Professor of Mathematical Modeling Dept., NRU MPEI, e-mail: abfrolov@mail.ru

Работа выполнена при поддержке: РФФИ (проект № 17-01-00485а)
The work is executed at support: RFBR (grants No. 17-01-00485a)

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов
Conflict of interests: the authors declare no conflict of interest

Статья поступила в редакцию: *15.03.2018*
The article received to the editor: *15.03.2018*