

Алгоритм построения мандатной политики безопасности для облачных систем

С.В. Белим
sbelim@mail.ru

С.Ю. Белим
belimsv@gmail.com

Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация

В статье предложен алгоритм реализации мандатного разграничения доступа на основе схемы предварительного распределения ключей. Необходимым условием является наличие иерархии субъектов. Использована модифицированная KDP-схема. Предложен метод построения семейства подмножеств, учитывающих подчиненность пользователей.

Введение

Основная проблема реализации различных моделей разграничения доступа в облачных системах состоит в отсутствии единого центра управления всеми вычислительными узлами. Для локальных систем принятие решения о разрешении или запрете доступа локализовано в одной службе, обрабатывающей запросы на доступ. Попытки построения облачных систем по аналогичной схеме противоречат самой идеологии облачных вычислений. В связи с чем необходимо использовать распределенные алгоритмы принятия решений о разграничении доступа.

На сегодняшний день наиболее высокий уровень безопасности обеспечивает мандатная политика разграничения доступа. Основным ее преимуществом является блокировка каналов утечки информации «по памяти». В рамках данной политики разграничения доступа всем субъектам и объектам системы присваиваются метки безопасности, характеризующие уровень доступа. Множество меток безопасности образует алгебраическую решетку. Для реализации в облачных системах необходимо организовать присвоение меток безопасности всем субъектам и объектам системы, а также алгоритм сравнения меток безопасности и принятия решения о доступе.

Данная задача может быть решена с помощью сертификатов открытых ключей. Однако при этом появляется выделенный сервер выдачи и подтверждения сертификатов. Такой сервер будет узким местом проектируемой системы. На который будут регулярно отправляться запросы, что приведет к существенному замедлению работы всей системы. Поэтому предпочтительным является решение, при котором единый центр распределения меток безопасности работает только на этапе инициализации системы. В процессе функционирования системы должны использоваться только распределенные алгоритмы без обращения к единому выделенному серверу.

Для распределения меток безопасности может быть использован один из алгоритмов предварительного распределения ключей шифрования, в которых осуществляется распределение ключевых материалов на начальном этапе инициализации пользователей и ресурсов. При дальнейшем функционировании системы связь осуществляется по зашифрованным каналам, ключи для которых вырабатываются на основе

Copyright © by the paper's authors. Copying permitted for private and academic purposes.

In: Sergey V. Belim, Nadezda F. Bogachenko (eds.): Proceedings of the Workshop on Data, Modeling and Security 2018 (DMS-2018),
Omsk, Russia, October 2018, published at <http://ceur-ws.org>

ключевых материалов и открытой информации. В этом случае необходимо модифицировать алгоритм формирования ключевых материалов, чтобы сделать невозможным шифрование для запрещенных каналов.

Классические схемы предварительного распределения ключей [1, 2] по-умолчанию предоставляют шифрованные каналы обмена информацией каждого абонента с каждым. Для учета требований по наличию запрещенных каналов необходима модификация этих схем. Модифицированная схема Блома для реализации дискреционной политики безопасности с единственным типом доступа предложена в статье [3]. Для систем с дискреционным разграничением доступа и симплексными каналами схема предварительного распределения ключей предложена в работе [4]. Модификация KDP-схемы для реализации дискреционной политики безопасности описана в статьях [5, 6]. Прямой перенос результатов, полученных для дискреционной политики безопасности, на системы с мандатной политикой безопасности невозможен, так как необходим учет иерархий субъектов и объектов. Построение ключевой схемы, учитывающей иерархию объектов с вычислимыми ключами предложено в работе [7]. Однако такая схема не позволяет полностью реализовать мандатную политику безопасности с обменом информацией между пользователями.

Целью данной статьи является модификация KDP-схемы предварительного распределения ключей с учетом иерархии пользователей для реализации мандатной политики безопасности в облачных системах.

1 Постановка задачи и схема распределения ключей

Пусть в системе авторизовано множество пользователей U . Иерархия пользователей на множестве U задается с помощью отношения порядка. Рассмотрим случай древовидной иерархии. Если пользователь u_i доминирует над пользователем u_j , то выполняется неравенство $u_i > u_j$. При древовидной иерархии также возможна ситуация, когда два элемента не сравнимы друг с другом, для этого случая введем обозначение $u_i <> u_j$. Сущность мандатной политики разграничения доступа состоит в том, чтобы разрешить информационные потоки только снизу–вверх по иерархии пользователей. То есть информационный поток от пользователя u_j к пользователю u_i разрешен тогда и только тогда, когда $u_i > u_j$. Все остальные информационные потоки запрещены. Исходя из этого можно сформулировать требования к схеме предварительного распределения ключей. Ключ для шифрования информации должен вырабатываться только в соответствии с иерхической подчиненностью. Для разрешенных информационных потоков ключи должны быть ненулевыми, а для запрещенных равны нулю.

Для реализации необходимой схемы предварительного распределения ключей модифицируем KDP-схему. В обычной KDP-схеме задается набор ключевых материалов $K = \{k_1, \dots, k_n\}$, который передается всем абонентам по защищенным каналам. Также определяется система $S = \{S_1, \dots, S_m\}$ множества $\{1, \dots, n\}$, где m – количество участников схемы. Множество S является открытым и хранится на общедоступном сервере. Для связи с пользователем u_j пользователь u_i запрашивает на сервере множества S_i и S_j . После этого пользователь u_i вычисляет пересечение множеств $S_{ij} = S_i \cap S_j$ и на основе полученного множества и набора ключевых материалов вычисляет парный ключ шифрования:

$$k_{ij} = \bigoplus_{l \in S_{ij}} k_l.$$

Пользователь u_j при получении запроса на установление канала обмена информацией, аналогично запрашивает у сервера множества S_i и S_j и по той же схеме вычисляет общий ключ шифрования.

Описанная KDP-схема не может быть использована напрямую, так как обеспечивает двусторонние каналы связи «каждый с каждым». Прежде всего, необходимо модифицировать схему таким образом, чтобы она формировала ключи несимметричные по перестановке индексов $k_{ij} \neq k_{ji}$. Как и в обычной схеме будем опираться на множество секретных ключевых материалов K и открытое множество S . Ключ для канала передачи информации от пользователя u_j к пользователю u_i будем вычислять по следующей формуле:

$$\Delta S_{ij} = \begin{cases} S_i \setminus S_j, & \text{if } S_i \cap S_j = \emptyset \\ S_i \cap S_j, & \text{otherwise} \end{cases}$$

$$k_{ij} = \bigoplus_{l \in \Delta S_{ij}} k_l.$$

При таком подходе требование несимметричности будет выполнено для всех ненулевых ключей. Пользователь u_i для чтения сообщений будет использовать ключи шифрования k_{ij} ($j = 1, \dots, m$), а для отправки

сообщений ключи k_{ji} ($j = 1, \dots, m$) Для запрещенных каналов передачи информации ключ шифрования должен быть нулевым:

$$k_{ij} = 0.$$

Данное условие будет выполнено только если пустым будет соответствующее множество:

$$\Delta S_{ij} = \emptyset.$$

Из этого равенства вытекает, что множество S не может быть произвольным, а должно быть согласовано с иерархической структурой пользователей.

Для реализации обычной KDP-схема используются семейства Шпернера. Семейством Шпернера [8] называется семейство подмножеств $D = \{D_1, \dots, D_n\}$ таких, что, если $D_i \cap D_j \subseteq D_t$, то либо $t = i$, либо $t = j$. Для формирования множества S используется некоторое семейство Шпернера D [9]. Сначала формируется семейство Шпернера D_i , элементы которого далее рассматриваются как пересечения подмножеств S_{ij} .

Построим множество S с учетом иерархии пользователей. Зададим семейство Шпернера с количеством элементов, равным числу пользователей $D = \{D_1, \dots, D_m\}$. Далее, совершая обход иерархического дерева пользователей, формируем множество S . Для пользователей $u_1, \dots, u_l$ (l —количество листовых вершин дерева) элементы множества S совпадают с соответствующими элементами семейства Шпернера:

$$S_i = D_i \quad (i = 1, \dots, l).$$

Для нелистовых вершин элементы множества S формируем на основе отношения подчиненности. Если пользователь u_i имеет ближайших потомков $u_{i1}, \dots, u_{ik}$, то соответствующий элемент множества вычисляется следующим образом:

$$S_i = S_{i1} \cup S_{i2} \cup \dots \cup S_{ik} \cup D_i.$$

Из построения видно, что при доминировании пользователя u_i над пользователем u_j будет выполняться неравенство $S_i \supset S_j$, следовательно $S_i \setminus S_j \neq \emptyset$, при этом $S_j \setminus S_i = \emptyset$. То есть ключ может быть сформирован только для разрешенного направления передачи информации. Для случая несравнимых вершин $u_i <> u_j$ ключ также будет иметь нулевое значение, так как $S_i \cap S_j = \emptyset$.

2 Пример построения схемы мандатного разграничения доступа

Рассмотрим пример реализации предложенной схемы для системы, включающей семь пользователей. Иерархия пользователей представлена на рисунке 1.

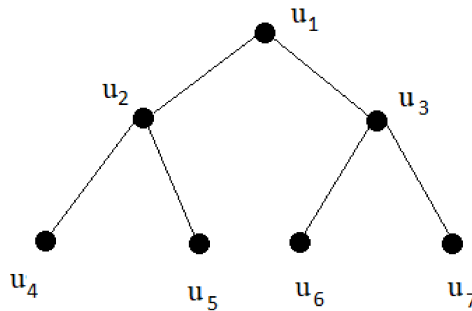


Рис. 1: Иерархия пользователей системы.

Пусть множество ключевых материалов включает в себя 15 однобайтовых элементов:

$$\begin{aligned}
 k_1 &= 00100100, \quad k_2 = 10101010, \quad k_3 = 01010101, \quad k_4 = 11011011, \\
 k_5 &= 11101110, \quad k_6 = 00010001, \quad k_7 = 10010010, \quad k_8 = 10110110, \\
 k_9 &= 00011000, \quad k_{10} = 11101110, \quad k_{11} = 10111001, \quad k_{12} = 11100111, \\
 k_{13} &= 00101101, \quad k_{14} = 11010010, \quad k_{15} = 01111111.
 \end{aligned}$$

Зададим семейство Шпернера следующим образом:

$$D_1 = \{1, 2\}, D_2 = \{3, 4\}, D_3 = \{5\}, D_4 = \{6, 7, 8\}, D_5 = \{9, 10\}, D_6 = \{11, 12, 13\}, D_7 = \{14, 15\}.$$

Элементы множества S определим следующим образом:

$$S_4 = D_4 = \{6, 7, 8\}, S_5 = D_5 = \{9, 10\}, S_6 = D_6 = \{11, 12, 13\}, S_7 = D_7 = \{14, 15\}.$$

Для каждого пользователя:

$$\begin{aligned} S_2 &= S_4 \cup S_5 \cup D_2 = \{3, 4, 6, 7, 8, 9, 10\}, \\ S_3 &= S_6 \cup S_7 \cup D_3 = \{5, 11, 12, 13, 14, 15\}, \\ S_1 &= S_2 \cup S_3 \cup D_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}. \end{aligned}$$

Процесс формирования множества S представлен на рисунке 2.

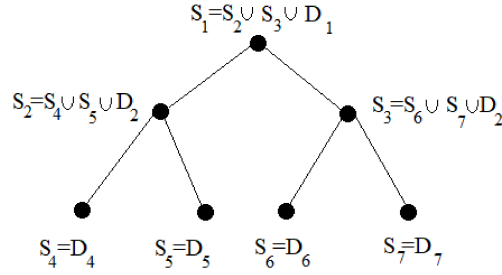


Рис. 2: Формирование множества S .

Для вычисления парных ключей используем разности множеств:

$$\begin{aligned} \Delta S_{12} &= S_1 \setminus S_2 = \{1, 2, 5, 11, 12, 13, 14, 15\}, \\ \Delta S_{13} &= S_1 \setminus S_3 = \{1, 2, 3, 4, 6, 7, 8, 9, 10\}, \\ \Delta S_{14} &= S_1 \setminus S_4 = \{1, 2, 3, 4, 9, 10, 11, 12, 13, 14, 15\}, \\ \Delta S_{15} &= S_1 \setminus S_5 = \{1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 14, 15\}, \\ \Delta S_{16} &= S_1 \setminus S_6 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 14, 15\}, \\ \Delta S_{17} &= S_1 \setminus S_7 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13\}, \\ \Delta S_{21} &= S_2 \setminus S_1 = \emptyset, \Delta S_{23} = S_2 \cap S_3 = \emptyset, \\ \Delta S_{24} &= S_2 \setminus S_4 = \{3, 4, 9, 10\}, \\ \Delta S_{25} &= S_2 \setminus S_5 = \{3, 4, 6, 7, 8\}, \\ \Delta S_{26} &= S_2 \cap S_6 = \emptyset, \Delta S_{27} = S_2 \cap S_7 = \emptyset, \\ \Delta S_{31} &= S_3 \setminus S_1 = \emptyset, \Delta S_{32} = S_3 \cap S_2 = \emptyset, \Delta S_{34} = S_3 \cap S_4 = \emptyset, \Delta S_{35} = S_3 \cap S_5 = \emptyset, \\ \Delta S_{36} &= S_3 \setminus S_6 = \{5, 14, 15\}, \\ \Delta S_{37} &= S_3 \setminus S_7 = \{5, 11, 12, 13\}, \\ \Delta S_{41} &= S_4 \setminus S_1 = \emptyset, \Delta S_{42} = S_4 \setminus S_2 = \emptyset, \Delta S_{43} = S_4 \cap S_3 = \emptyset, \Delta S_{45} = S_4 \cap S_5 = \emptyset, \\ \Delta S_{46} &= S_4 \cap S_6 = \emptyset, \Delta S_{47} = S_4 \cap S_7 = \emptyset, \\ \Delta S_{51} &= S_5 \setminus S_1 = \emptyset, \Delta S_{52} = S_5 \setminus S_2 = \emptyset, \Delta S_{53} = S_5 \cap S_3 = \emptyset, \Delta S_{54} = S_5 \cap S_4 = \emptyset, \\ \Delta S_{56} &= S_5 \cap S_6 = \emptyset, \Delta S_{57} = S_5 \cap S_7 = \emptyset, \\ \Delta S_{61} &= S_6 \setminus S_1 = \emptyset, \Delta S_{62} = S_6 \cap S_2 = \emptyset, \Delta S_{63} = S_6 \setminus S_3 = \emptyset, \Delta S_{64} = S_6 \cap S_4 = \emptyset, \\ \Delta S_{65} &= S_6 \cap S_5 = \emptyset, \Delta S_{67} = S_6 \cap S_7 = \emptyset, \\ \Delta S_{71} &= S_7 \setminus S_1 = \emptyset, \Delta S_{72} = S_7 \cap S_2 = \emptyset, \Delta S_{73} = S_7 \setminus S_3 = \emptyset, \Delta S_{74} = S_7 \cap S_4 = \emptyset, \\ \Delta S_{75} &= S_7 \cap S_5 = \emptyset, \Delta S_{76} = S_7 \cap S_6 = \emptyset. \end{aligned}$$

Парные ключи определяются из равенств:

$$\begin{aligned}
k_{12} &= k_1 \oplus k_2 \oplus k_5 \oplus k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} = 10111110, \\
k_{13} &= k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_6 \oplus k_7 \oplus k_8 \oplus k_9 \oplus k_{10} = 11000011, \\
k_{14} &= k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} = 00101000, \\
k_{15} &= k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6 \oplus k_7 \oplus k_8 \oplus k_{11} \oplus k_{12} \oplus k_{13} \oplus k_{14} \oplus k_{15} = 00000101, \\
k_{16} &= k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6 \oplus k_7 \oplus k_8 \oplus k_9 \oplus k_{10} \oplus k_{14} \oplus k_{15} = 10000000, \\
k_{17} &= k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6 \oplus k_7 \oplus k_8 \oplus k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12} \oplus k_{13} = 01011110, \\
k_{21} &= 0, k_{23} = 0, \\
k_{24} &= k_3 \oplus k_4 \oplus k_9 \oplus k_{10} = 01111000, \\
k_{25} &= k_3 \oplus k_4 \oplus k_6 \oplus k_7 \oplus k_8 = 10111011, \\
k_{26} &= 0, k_{27} = 0, \\
k_{31} &= 0, k_{32} = 0, k_{34} = 0, k_{35} = 0, \\
k_{36} &= k_5 \oplus k_{14} \oplus k_{15} = 01100011, \\
k_{37} &= k_5 \oplus k_{11} \oplus k_{12} \oplus k_{13} = 10011111, \\
k_{41} &= 0, k_{42} = 0, k_{43} = 0, k_{45} = 0, k_{46} = 0, k_{47} = 0, \\
k_{51} &= 0, k_{52} = 0, k_{53} = 0, k_{54} = 0, k_{56} = 0, k_{57} = 0, \\
k_{61} &= 0, k_{62} = 0, k_{63} = 0, k_{64} = 0, k_{65} = 0, k_{67} = 0, \\
k_{71} &= 0, k_{72} = 0, k_{73} = 0, k_{74} = 0, k_{75} = 0, k_{76} = 0.
\end{aligned}$$

Схема предварительного распределения ключей, удовлетворяющая иерархии субъектов, представлена на рисунке 2. Разрешенными являются только каналы «снизу – вверх». Несравнимые пользователи не взаимодействуют. Рассмотрим иерархию субъектов, отличную от древовидной. Пример представлен на рисунке 3.

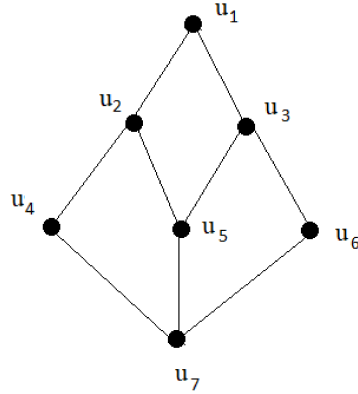


Рис. 3: Иерархия пользователей отличная от древовидной.

По такому же принципу могут быть вычислены элементы множества S (рисунок 4). Ключи, запрещенные мандатной политикой безопасности будут нулевыми.

Заключение

Схема, предложенная в данной статье, позволяет реализовать предварительное распределение ключей симметричного шифрования для распределенных систем с иерархией пользователей. Как и в случае KDP-схемы используются семейства Шпернера. Однако обычная KDP-схема позволяет формировать двунаправленные каналы обмена информацией, тогда как предложенная схема ориентирована на симплексные

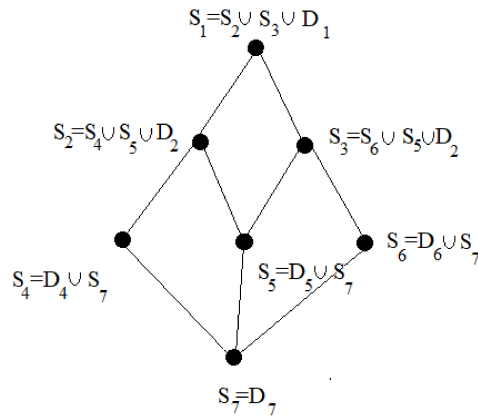


Рис. 4: Вычисление элементов множества S .

каналы обмена информацией. Следует отметить, что предложенная модификация схемы предварительного распределения ключей не ведет к росту объема ключевых материалов, что является обязательным условием использования данного подхода к построению защищенных систем.

Список литературы

- [1] R. Blom. An optimal class of symmetric key generation systems. *Proc. of the EUROCRYPT 84 workshop on Advances in cryptology: theory and application of cryptographic techniques.*:335–338, 1985.
- [2] C.J. Mitchell, C. Piper. Key storage in Secure Networks. *Discrete and Applied Math.*, 21:215–228, 1988.
- [3] S.V. Belim, S.Yu. Belim, S.Yu. Polyakov The Implementation of Discretionary Access Separation Using a Modified Blom's Scheme of Key Distribution. *Information Security Problems. Computer Systems.*, 3:72–76, 2015.
- [4] S.V. Belim, S.Yu. Belim. The Modification of Blom's Key Predistribution Scheme, Taking into Account Simplex Channels. *Information Security Problems. Computer Systems.*, 3:82–86, 2017.
- [5] S.V. Belim, S.Yu. Belim. KDP Scheme of Preliminary Key Distribution in Discretionary Security Policy. *Automatic Control and Computer Sciences.*, 50(8):777–786, 2016.
- [6] S.V. Belim, S.Yu. Belim. The VPN Implementation on Base of the KDP-Scheme. *CEUR Workshop Proceedings.*, 1732, 2016. URL: <http://ceur-ws.org/Vol-1732/paper3.pdf>
- [7] S.V. Belim, N.F. Bogachenko. Distribution of Cryptographic Keys in Systems with a Hierarchy of Objects. *Automatic Control and Computer Sciences.*, 50(8):773–776, 2016.
- [8] M. Dyer, T. Fenner, A. Frieze, A. Thomason. On key storage in secure networks. *J. Cryptology.*, 8:189–200, 1995.
- [9] C.M. O'Keefe. Applications за finite geometries in information security. *Australas. J. Combin.*, 7:195–212, 1993.

Algorithm of Mandatory Security Policy Creation for Cloudy Systems

Sergey V. Belim Svetlana Yu. Belim

In article the algorithm of mandatory access control using the scheme of keys preliminary distribution is developed. Necessary condition is existence the subjects hierarchy. The modified KDP-scheme is used. The method for creation the subsets family considering subordination the users hierarchy is developed.