Реализация VPN на основе KDP-схемы

С.В. Белим
belimsv@omsu.ru

С.Ю. Белим
sbelim@mail.ru

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

Аннотация

В статье предложена модификация KDP-схемы. Реализована дискреционная политика безопасности на основе KDP-схемы. Эта схема используется для реализации VPN. Предложен алгоритм вычисления пар ключей.

Введение

Под попарной ключевой схемой предварительного распределения ключей принято понимать схему, в которой один ключ ассоциирован ровно с двумя узлами. В примитивной схеме каждый узел хранит $(n-1)$ ключ для каждого узла сети. Эта схема устойчива к атакам злоумышленника, но обладает плохой масштабируемостью и требует больших ресурсов памяти для хранения ключевых материалов.

Все схемы предварительного распределения ключей подразумевают возможность связи каждого абонента сети с каждым. Тогда как в реальных системах существуют политики безопасности, ограничивающие возможности взаимодействия отдельных пар пользователей. Базовой политикой безопасности является дискреционное разделение доступа, заданное в виде матрицы доступов, определяющих разрешенные каналы передачи информации.

Одной из самых надежных схем, хорошо разработанной и ставшей уже классической, является KDP-схема [1, 2], основанная на построении семейства пересекающихся множеств.

Схема предварительного распределения ключей на основе многомерных пространств (DDHV-схема) предложена в работе [3]. В данной схеме вместо одного глобального ключевого пространства используется набор ключевых пространств. Каждому узлу сопоставляется подмножество ключевых пространств. Для каждого отдельного ключевого пространства используется схема Блома. Вероятность того, что два узла смогут выработать совместный ключ равна вероятности того, что они имеют хотя бы одно общее ключевое пространство. DDHV-схема абсолютно устойчива, если количество скомпрометированных узлов ниже некоторого порогового значения.

Схема обмена ключами, основанная на модели злоумышленника представлена в работе [4]. Данная схема основывается на предположении о том, что злоумышленник не может контролировать все линии связи. Вводится вероятность прослушивания линии связи злоумышленником. После чего исследуется возможность незащищенного обмена ключами между соседними узлами. Злоумышленник сможет атаковать систему, только если будет прослушивать канал связи в момент передачи ключевой информации.

В связи с развитием беспроводных систем связи широкое распространение получили вероятностные схемы предварительного распределения ключей, обзор которых представлен в работе [5].

В статье [6] представлены две схемы, которые основаны на некоторой информации о процедуре развертывания сети. В первой схеме попарный ключ генерируется только для тех пар узлов, для которых высока вероятность оказаться физическими соседями при разворачивании сети. Второй подход основан

In: Sergey V. Belim, Nadezda F. Bogachenko (eds.): Proceedings of the Workshop on Data Analysis and Modelling (DAM 2016), Omsk, Russia, October 2016, published at http://ceur-ws.org

на пороговой схеме, использующей многочлены от двух переменных. Область развертывания делится на сектора подходящего размера. Для каждого сектора используется свое ключевое пространство. Ключи генерируются только для соседей из близких секторов.

В статье [7] предложена схема, в которой область развертывания сети разделена на сектора, соответствующие некоторой группе узлов. Для каждой группы существует ключевой пул, созданный таким образом, что существуют перекрытия с ключевыми пулами для соседних групп. Совместное использование некоторого подмножества ключей возможно только для групп с прямыми соседними секторами.

Аналогичный подход, в котором исходная группа всех узлов сети делится на меньшие подгруппы, представлен в работе [8]. Узлы, которые совместно решают одну задачу должны иметь возможность связаться, поэтому они относятся к одной подгруппе.

В статье [9] проведено исследование схем распределения ключей, основанных на предположении о случайном появлении узла и делается вывод о низкой устойчивости таких схем. Для повышения устойчивости схемы распределения ключей предложен выборочный алгоритм получения узла. Новый подход также основан на разделении всей сети на зоны на этапе развертывания. Проведено исследование предложенной схемы.

В статье [10] предложен подход также использующий разделение сети на кластеры. При этом вводится два типа кластеров - контролируемые и неконтролируемые. Контролируемые кластеры предназначены для обеспечения безопасности критических частей сети и содержат более мощный узел супервизора. Попарные ключи вычисляются для связи между супервизором и остальными узлами сети. В неконтролируемой части используется один общий ключ для всех узлов кластера.

На сегодняшний день широкое распространение получили технологии построения виртуальных сетей, логическая топология которых не совпадает с физической. Рассмотрим возможность построения виртуальной сети с помощью алгоритмов предварительного распределения ключей. Пусть задана некоторая вычислительная сеть, которая может быть, как локальной, так и глобальной. Виртуальная сеть задается в виде логических связей между реальными узлами сети. Будем рассматривать топологию виртуальной сети как неориентированный граф. Множество связей сети является множеством ребер графа и задается матрицей инцидентности, в которой единица соответствует наличию ребра, а нуль - его отсутствию. По своему смыслу наличие логической связи означает возможность прямого информационного обмена между узлами. Отсутствие связи означает запрет на прямую связь двух узлов. Следовательно мы имеем множество узлов и набор правил, разграничивающих возможности обмена информацией между ними. Данная конструкция может быть реализована с помощью матрицы доступов в рамках дискреционной политики безопасности. Очевидно, что матрица доступов будет совпадать с матрицей инцидентности, графа определяющего топологию виртуальной сети.

В общем случае постановка задачи выглядит следующим образом. В некотором виде задана топология сети в виде множества вершин и множества связей между ними. Требуется с помощью механизма распределения ключей разграничить возможность взаимодействия узлов сети таким образом, чтобы были возможны только связи, разрешенные топологией сети.

Неотъемлемой составляющей системы на основе данного подхода является сервер распределения ключей, который может быть реализован как выделенный узел сети. Однако такое решение является неудобным, так как ведет к росту размера сети и дополнительным линиям связи. Более оправданным будет размещение сервера распределения ключей в качестве сервиса на одном из существующих узлов сети. Такое техническое решение не приведет к значительному увеличению нагрузки на данный узел, так как сервер распределения ключей выполняет достаточно небольшой объем операций.

Следует отметить, что построенная сеть обладает свойствами виртуальной частной сети (VPN), так как все сообщения передаются в зашифрованном виде. Для того, чтобы предложенная сеть стала полноправной VPN необходимо добавить в систему протоколы с открытым ключом для передачи ключевых материалов по открытым каналам. Например, может быть использован протокол Диффи-Хеллмана для установления соединения и передачи ключевых материалов между сервером распределения ключей и узлами сети. Для повышения надежности системы ключ, вырабатываемый на основе ключевых материалов, может быть использован только на этапе установления соединения для передачи сеансового ключа. Также ключевые материалы могут быть использованы для аутентификации узлов сети.

1 Модификация KDP-схемы с учетом запрещенных каналов

Пусть в системе имеется n пользователей u_1, ..., u_n, между которыми необходимо организовать защищенный обмен информации на основе симметричной криптографии. Рассмотрим систему, в которой задано дискреционное разделение доступа, запрещающее обмен сообщениями между некоторыми парами пользователей. Представим ограничения на каналы связи в виде матрицы M, в которой пользователю u_i соответствует i-ая строка и i-ый столбец. Ячейка матрицы $M[i,j] = 1$, если разрешен обмен информацией между i-ым и j-ым пользователем, и $M[i,j] = 0$, если обмен запрещен. Задача сводится к тому, что необходимо построить схему предварительного распределения ключей таким образом, чтобы она вырабатывала общий ключ, если обмен разрешен и не позволяла получить ключ, если обмен запрещен.

Будем решать поставленную задачу с помощью модификации классической KDP-схемы [1, 2]. В традиционной KDP-схеме задается множество ключей $K = \{k_1, ..., k_m\}$, которые передаются всем участникам защищенной сети заранее по защищенному каналу и держатся в секрете. Все ключи пронумерованы. После этого строится семейство подмножеств $S = \{S_1, ..., S_n\}$ множества $\{1, 2, ..., n\}$. Каждому пользователю u_i сопоставляется подмножество S_i. Семейство S является открытым. Для получения общего ключа шифрования пользователи u_i и u_j должны извлечь из открытой базы подмножества S_i и S_j и найти их пересечение $S_{ij} = S_i \cap S_j$. Общий ключ k_{ij} определяется на основе подмножества ключей k_l, где $l \in S_{ij}$, по какому-либо преобразованию, известному всем участникам защищенной сети.

Традиционная KDP-схема строится на основе семейств Шпернера. Семейством Шпернера [][12] называется семейство подмножеств $D = \{D_1, ..., D_n\}$ таких, что, если $D_i \cap D_j \subseteq D_t$, то либо $t = i$, либо $t = j$. Семейство множеств S строится на основе семейства Шпернера D [11]. Элементы семейства Шпернера D_i используются в качестве пересечений подмножеств S_{ij}.

Для построения KDP-схемы с учетом дискреционного разделения доступа потребуем, чтобы для пары пользователей u_i и u_j, которым запрещен обмен информацией, подмножество возможных ключей было нулевым. Другими словами, если $M[i,j] = 0$, то $S_{ij} = \emptyset$, а если $M[i,j] = 1$, то $S_{ij} \neq \emptyset$.

Рассмотрим возможность реализации новых требований к KDP-схеме. Как и при построении традиционной KDP-схемы будем использовать семейство Шпернера дополненное некоторым количеством пустых множеств. Пусть в матрице $M[i,j]$ выше главной диагонали расположено m единичных элементов. Можно ограничиться рассмотрением элементов только выше главной диагонали, так как по постановке задачи каналы обмена информацией являются симметричными, а, следовательно, матрица $M[i,j]$ является симметричной. На основе множества $\{1, 2, ..., n\}$ строим семейство Шпернера, содержащее m элементов – $D = \{D_1, ..., D_m\}$. Введем матрицу $MD[i,j]$, элементами которой являются множества. Заполним матрицу $MD[i,j]$ следующим образом. Если $M[i,j] = 1$, то $MD[i,j] = D_k$ и $MD[j,i] = D_k$. Подмножество из семейства Шпернера выбирается случайным образом. Каждое подмножество D_k используется только один раз для инициализации двух элементов матрицы $MD[i,j]$, симметричных относительно главной диагонали. Если $M[i,j] = 0$, то $MD[i,j] = \emptyset$ и $MD[j,i] = \emptyset$. Множества S_i найдем как объединение подмножеств из семейства Шпернера, расположенных в i-ой строке.

$$S_i = \bigcup_{j=1}^{n} MD[i,j] \quad (i = 1, ..., n).$$

Очевидно, что в силу симметричности матрицы $MD[i,j]$ объединение множеств можно проводить как по строке, так и по столбцу.

2 Алгоритм построения VPN на основе KDP-схемы

Рассмотрим возможность построения виртуальной сети с помощью KDP-схемы. На вход системы подается список узлов сети LU и список пар узлов, между которыми существуют виртуальные соединения. Алгоритм имеет следующий вид:

1. Получаем матрицу инцидентности графа, определяющей топологию сети, которая в дальнейшем играет роль матрицы доступов.

2. Создаем ключевое множество $K = \{k_i\}_{i=1,...,n}$. Мощность множества ключей n должна быть не менее максимально возможного количества связей

$$n > \frac{(|LU|(|LU| - 1))}{2}.$$

Элементы ключевого множества k_i $(i = 1, ..., n)$ задаются с помощью генератора псевдослучайной последовательности.

3. Определяем количество единичных элементов B матрицы M, расположенных выше главной диагонали.

4. Разбиваем множество $N = \{1, ..., n\}$ на B не пересекающихся подмножеств D_1,..., D_B, так чтобы

$$\bigcup_{i=1}^{B} D_i = N.$$

5. Строим матрицу MD, распределяя случайным образом множества D_i по ячейкам матрицы MD, соответствующих единичным элементам матрицы M.

6. Формируем множества S_i, как объединение множеств D_j расположенных в одной строке. Полученные множества S_i, хранящиеся в открытом виде, вместе с множеством ключей K, хранящемся в секрете, и будут играть роль ключевых материалов.

При формировании таблицы маршрутизации, для определения наличия связи между узлами i и j, системе необходимо определить пересечение множеств $S_i \cap S_j$ и, если оно окажется не пустым, то узлы i и j связаны. Парный ключ будет вычисляться как побитовая операция XOR между всеми элементами множества ключей, имеющих номера из множества $S_i \cap S_j$.

В пункте 4 алгоритма необходимо случайным образом разбить множество на не пересекающиеся подмножества. Для автоматизации процесса разбиение будем производить с помощью линейного конгруэнтного генератора псевдослучайных последовательностей. Генератор нужен для того чтобы определить какой элемент отнести к какому подмножеству. Если на i-ом шаге будет получено число x_i, то элемент множества i отнесем к подмножеству с номером x_i (Dx_i). Очевидно, что числа x_i должны лежать в интервале от 1 до B. То есть линейный конгруэнтный генератор должен быть по модулю B. Однако не для каждого B можно построить генератор с полным периодом. Поэтому мы немного ухудшим характеристики генератора, но при этом будем гарантировать, что в псевдослучайной последовательности будут встречаться все числа от 1 до B, и не будет других чисел.

Выберем минимальное целое число p, превышающее B. Вычисляем псевдослучайную последовательность y_i с помощью линейного конгруэнтного генератора:

$$y_i = ay_i + b \ (mod \ p),$$

где $b \neq 0$, y_0 – произвольное целое число, a – примитивный элемент в кольце Z_p. После этого вычисляем псевдослучайную последовательность x_i по формуле:

$$x_i = y_i \ (mod \ B) + 1.$$

Исходя из того, что по построению a – примитивный элемент в кольце Z_p, то псевдослучайная последовательность y_i будет иметь максимальный период, то есть в ней будут встречаться все числа от 0 до $p-1$, а, следовательно, в последовательности x_i будут встречаться все числа от 1 до B. К множеству D_j отнесем те числа z, для которых $x_z = j$. Очевидно, что полученный набор подмножеств будет семейством Шпернера, так как номер множества, к которому будет отнесен элемент, определяется однозначно.

Для распределения множеств D_i по ячейкам матрицы MD упорядочим ячейки матрицы M, лежащие выше главной диагонали и содержащие единичные элементы, по возрастанию по сумме индексов. Если два элемента имеют одинаковую сумму индексов, то первым будет идти тот, у которого меньше номер строки. Сопоставим этим ячейкам множества семейства Шпернера, упорядоченные по возрастанию их индекса. Внесем соответствующее множество из семейства Шпернера в ячейки матрицы MD, с индексами, совпадающими с индексами элемента матрицы M и перестановке индексов. Дополнительное перемешивание подмножеств D_i не требуется, так как они сформированы достаточно случайно.

Заключение

Таким образом предложенная модифицированная KDP-схема предварительного распределения ключей позволяет реализовать виртуальные компьютерные сети. Виртуальные сети, реализованные с помощью схемы предварительного распределения ключей обладают всеми необходимыми атрибутами для функционирования в качестве виртуальных частных сетей. Для полнофункциональной реализации виртуальной частной сети необходимо схемы предварительного распределения ключей дополнить криптографическими протоколами на основе криптографии с открытым ключом.

Список литературы

[1] C.J. Mitchell, C. Piper, Key storage in Secure Networks. *Discrete and Applied Math.* 21:215–228, 1988.

[2] C.J. Mitchell, Combinatorial techniques for key storage reduction in secure networks. *Technical memo.* Hewlett-Packard Laboratories. Bristol, 1988.

[3] W. Du, J. Deng, Y.S. Han, P.K. Varshney, A pairwise key pre-distribution for wireless sensor networks. *CCS03, Washington, DC, USA.* 42–51, 2003.

[4] R. Anderson, H. Chan, A. Perrig Key infection: Smart trust for smart dust. *ICNP'04, Berlin, Germany.* 2004.

[5] H. Chan, A. Perrig, D. Song, Key distribution techniques for sensor networks. *in Wireless Sensor Networks, Springer US.* 277–303, 2004.

[6] D. Liu, P. Ning, Location-based pairwise key establishments for static sensor networks. *1st ACM Workshop Security of Ad Hoc and Sensor Networks Fairfax, Virginia.* 72–82, 2003.

[7] W. Du, J. Deng, Y.S. Han, P.K. Varshney, A key management scheme for wireless sensor networks using deployment knowledge. *IEEE Transactions on Parallel & Distributed Systems.* 19(10):1411–1425, 2008.

[8] Y. Jin, L. Wang, Y. Kim, X.-Z. Yang, Energy Efficient Non-uniform Clustering Division Scheme in Wireless Sensor Networks. *Wireless Personal Communications.* 45(1):31–43, 2008.

[9] D. Huang, M. Mehta, D. Medhi, L. Harn, Location-aware key management scheme for wireless sensor networks. *SASN04, Washington, DC, USA.* 29–42, 2004.

[10] Y.W. Law, R. Corin, S. Etalle, P.H. Hartel, A formally verifed decentralized key management architecture for wireless sensor networks. *in Personal Wireless Communications.* 27–39, Springer Berlin Heidelberg, 2003.

[11] M. Dyer, T. Fenner, A. Frieze, A. Thomason, On key storage in secure networks. *J. Cryptology*, 8:189–200, 1995.

The VPN Implementation on Base of the KDP-Scheme

Sergey V. Belim, Svetlana Yu. Belim

In paper the modification of KDP-scheme is suggested. The discretionary security policy is implemented on base of KDP-scheme. This scheme is used for release VPN. The algorithm for pair keys calculation is developed.