УДК 004

ШИФРОВАНИЕ НА ОСНОВЕ АТРИБУТОВ

Шабанин И.О.

Магистрант кафедры информационной безопасности,

Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)

г. Москва, Россия

Аннотация: Цель работы – продемонстрировать принцип работы криптосистем на основе атрибутов. В работе были описаны задачи, решаемые с помощью данной технологии. Рассмотрены две основные схемы: атрибутная схема шифрования с правилом доступа на основе ключа и атрибутная схема шифрования с правилом доступа на основе шифртекста. Проведен разбор этапов построения подобных криптосистем.

Ключевые слова: атрибутная схема, шифрование, криптосистема, информация.

ATTRIBUTE-BASED ENCRYPTION

Shabanin I.O.

Master student of the department of information Security,

Moscow State Technical University named after N.E. Bauman (National Research University)

Moscow, Russia

Abstract: the Purpose of this paper is to demonstrate the principle of operation of attribute-based cryptosystems. The paper describes the problems solved by this technology. Consider two main schemes: an attribute encryption scheme with a key-based access rule and an attribute encryption scheme with a ciphertext-based access rule. The analysis of the stages of construction of such cryptosystems was carried out.

Keywords: attribute scheme, encryption, cryptosystem, information.

В настоящее время облачные вычисления получили стремительное развитие. Эта технология в действительности представляет интерес для пользователей благодаря своей возможности предоставления доступа к данным в любой момент времени из любого места. Также, ресурсы, используемые в облачных сервисах, в значительной мере превосходят ресурсы пользователей, поэтому, с их помощью можно решать более серьезные задачи за более короткое время. Но, на ряду с этим, облачная инфраструктура представляет повышенные риски и более ограниченную возможность контроля. В этом заключаются главные проблемы облачных вычислений – защита информации и доверие пользователей по отношению к облачным провайдерам, аутентификация участников информационного обмена. Однако эти проблемы могут быть решены с помощью использования криптографических методов.

Для аутентификации пользователей используется криптография с открытым ключом. В таких криптосистемах каждый пользователь имеет свой закрытый ключ и связанный с ним открытый ключ. Удостоверяющий центр (УЦ) – объект, которому доверяют все пользователи, гарантирует подлинность открытых ключей. Для этого УЦ для каждого открытого ключа выпускает сертификат. Сертификат содержит открытый ключ пользователя и идентифицирующую этого пользователя информацию, а также другую служебную информацию. Сертификат заверяется электронной цифровой подписью (ЭЦП) УЦ. Получаемый объект называется сертификатом открытого ключа пользователя. Таким образом, криптосистемам, которые используют инфраструктуру открытых ключей, необходима система управления цифровыми сертификатами, которая, как правило, слишком сложна в обслуживании и функционировании [1].

Чтобы уменьшить сложность, присущую традиционным асимметричным криптосистемам, из-за наличия системы управления цифровыми сертификатами Ади Шамир предложил схему шифрования, основанную на идентификаторах

(ID-based encryption, IBE) [2]. Шифрование на основе атрибутов можно рассматривать как обобщение шифрования, основанного на идентификаторах, вскоре идея IBE была улучшена в работе Амита Сахаи и Брента Уотерса. Амит Сахаи и Брент Уотерс ввели понятие шифрования на основе атрибутов (Attribute based encryption, ABE) [3].

Шифрование на основе атрибутов можно рассматривать как обобщение шифрования, основанного на идентификаторах. Однако в отличие от IBE, криптосистемы на основе атрибутного шифрования могут так же применяться и для контроля доступа к шифрованным данным.

В криптосистемах на основе атрибутов закрытый ключ пользователя генерируется и выпускается доверенным центром. Этот закрытый ключ генерируется на основе некоторого набора атрибутов. Владелец данных шифрует данные при помощи открытого ключа и некоторого набора атрибутов. При расшифровании шифртекста проверяется соответствие между атрибутами, которые составляют закрытый ключ пользователя, и атрибутами зашифрованных данных. Если число совпадающих атрибутов превышает некоторый установленный порог d, то пользователь сможет корректно расшифровать данные.

Пусть $A = A_1, A_2, \dots, A_n \quad (1) -$ множество атрибутов.

Набор $\Lambda \subseteq 2^{\{A=A_1,A_2,\dots,A_n\}}$ (2) назовем монотонным,

если $\forall B, C: B \in A, B \subseteq C \Rightarrow C \in \Lambda$. (3) структура доступа – непустой монотонный набор $\Lambda \subseteq 2^{\{A=A_1,A_2,\dots,A_n\}}$.

Набор атрибутов, который принадлежат множеству Λ назовем авторизованным набором атрибутов. Набор атрибутов, который не принадлежит множеству Λ назовем неавторизованным набором атрибутов.

Далее под структурой доступа будем понимать монотонную структуру доступа. Стоит заметить, что в 2007 году авторы ABE-схемы предложили обобщенную схему на случай немонотонных структур доступа, однако в 2013 году группа ученых из Тайваня в работе [4] показали, что такая схема является неэффективной.

Общая схема шифрования на основе атрибутов состоит из четырёх алгоритмов: Инициализации (Setup), генерации закрытого ключа пользователя (KeyGen), шифрования (Encrypt), и расшифрования (Decrypt):

Инициализация

$Setup()$: Доверенный центр случайным образом выбирает элементы $t_i \in Z_p$ и случайный элемент $y \in Z_p, i = \overline{1,n}.$ (4) Тогда открытый ключ PK и мастер ключ MK:

$$PK = (T_1 = g^{t_1}, \dots, T_n = g^{t_n}, Y = \varphi(g,g)^y, \quad (5)$$

$$MK = (t_1, \dots, t_n, y) \quad (6)$$

Генерация закрытого ключа пользователя

$KeyGen(\tau, MK)$: Алгоритм возвращает ключ, с помощью которого пользователь сможет расшифровать сообщение, зашифрованное с использованием атрибутов A. Генерация закрытого ключа осуществляется по следующему правилу. Для каждого узла x дерева τ задаётся многочлен q_x степени $d_x = k_x -$

1, причем $q_x(0) = q_{parant(x)}\big(index(x)\big)$. (7) Для корневого узла задается полином такой, что $q_r(0) = y$. Затем случайно выбирается d_r значений для того, чтобы полностью определить многочлен q_r. Тогда закрытый ключ пользователя:

$$D = \left\{D_i = g^{\frac{q(i)}{t_i}}\right\}, \text{где } i = att(x) \quad (8)$$

Шифрование

$Encrypt(A, PK, M)$: Пользователь шифрует сообщение $M \in G_2$ с использованием набора атрибутов A и случайно выбранного числа $s \in Z_p$, тогда шифртекст:

$$CT = (A, E = MY^s, \{E_i = T_i^s\}_{i \in A}) \quad (9)$$

Расшифрование

$Decrypt(CT, D)$: Пользователь расшифровывает шифртекст CT, используя свой закрытый ключ D. Определим рекурсивный алгоритм

$$Decrypt(CT, D) = \varphi(D_x, E_i) = \varphi\left(g^{\frac{q_x(0)}{t_i}}, g^{st_i}\right) = \varphi(g, g)^{sq_x(0)} \quad (10)$$

Тогда для корневого узла r имеем

$$DecryptNode(CT, D, r) = \varphi(g, g)^{ys} = Y^s, \quad (11)$$

Тогда исходное сообщение

$$M = \frac{E}{Y^s}. \quad (12)$$

Атрибутная схема шифрования с правилом доступа на основе ключа была предложена Гоалом (Goyal) в 2006 году в работе [5]. В предложенной схеме каждый шифртекст помечается некоторым набором описательных атрибутов, а в закрытом ключе пользователя содержится правило доступа к шифрованным данным. Данные могут быть расшифрованы только тогда, когда набор атрибутов данных соответствует структуре доступа в закрытом ключе пользователя. Структура доступа формируется в виде условий, принимающих при проверке истинное или ложное значение и соединяемых с помощью логических операций «И» и «ИЛИ».

Атрибутная схема шифрования с правилом доступа на основе ключа, как и классическая схема ABE-шифрования, состоит из 4-х алгоритмов: инициализации (генерация открытого ключа и универсального ключа), генерации закрытого ключа пользователя, шифрования и расшифрования. Рассмотри схему подробнее.

Введем некоторые обозначения. A_{u-KP} – структура доступа в закрытом ключе пользователя. A_{CT} – атрибуты, используемые для шифрования данных.

Инициализация

$Setup()$: Доверенный центр случайным образом выбирает элементы $t_i \in Z_p$ и случайный элемент $y \in Z_p, i = \overline{1,n}$. Тогда открытый ключ PK и мастер ключ MK:

$$PK = (T_1 = g^{t_1}, \dots, T_n = g^{t_n}, Y = \varphi(g,g)^y) \quad (13)$$

$$MK = (t_1, \dots, t_n, y) \quad (14)$$

Генерация закрытого ключа пользователя

$KeyGen(A_{u-KP}, PK, MK)$: Третья доверенная сторона генерирует компоненты закрытого ключа для каждого узла x, тогда закрытый ключ:

$$D = \{D_x = g^{\frac{q_x(0)}{t_i}}\} \quad (14)$$

где i равен узлу листа в структуре доступа.

Шифрование

$Encrypt(A_{CT}, PK, M)$: Пользователь шифрует сообщение $M \in G_2$ с использованием набора атрибутов A_{CT} и случайно выбранного числа $s \in Z_p$, тогда шифртекст:

$$CT = (A_{CT} = MY^s, \{E_i = T_i^s\}_{i \in A_{CT}}) \quad (15)$$

Расшифрование

$Decrypt(CT, D)$: Данный алгоритм является рекурсивным. На вход алгоритму поступает шифртекст, секретный ключ пользователя, набор атрибутов, которые ассоциированы с секретным ключом пользователя. Определим рекурсивный алгоритм

$$DecryptNode(CT, D, x) = \varphi(D_x, E_i) = \varphi\left(g^{\frac{q_x(0)}{t_i}}, g^{st_i}\right) = \varphi(g, g)^{sq_x(0)} \quad (16)$$

Пусть i – лист и он содержится в структуре доступа секретного ключа, тогда применяется алгоритм $DecryptNode$. Если i – узел, то алгорит $DecryptNode$ применяется для наследников узла. Последний этап – применение алгоритма $DecryptNode$ для корня дерева, то есть вычисляется $\varphi(g, g)^{sy} = Y^s$.

Таким образом, исходное сообщение M может быть получено следующим образом:

$$M = \frac{E}{Y^s} (17)$$

Атрибутная схема шифрования с правилом доступа на основе шифртекста по принципу работы схожа со схемой шифрования с политикой, привязанной к ключу. В данной схеме у каждого объекта доступа, например, файла собственная политика правил доступа, а субъекты доступа, то есть пользователи, помечаются атрибутами. В обеих схемах принцип основан на проверке условия, удовлетворяют ли атрибуты политике правил доступа и наоборот. По гибкости задания прав доступа и по количеству объектов контроля доступа представленные схемы не отличаются друг от друга. Привязка политики доступа к шифртекстам или к ключам не имеет существенного значения с точки зрения возможностей для формулирования правил доступа, поэтому решение о реализации той или иной схемы должно приниматься, в основном, из соображений удобства.

Данная схема, как и классическая схема ABE-шифрования, состоит из 4-х алгоритмов: инициализации (генерация открытого ключа и универсального ключа), генерации закрытого ключа пользователя, шифрования и расшифрования. В дополнение предоставляется пятый алгоритм – делегат (delegat).

Инициализация.

Не принимает входных параметров, кроме неявного параметра безопасности. Алгоритм возвращает открытый ключ PK и мастер ключ MK.

Генерация закрытого ключа пользователя.

Алгоритм генерации ключа принимает на вход мастер ключ MK и набор атрибутов S, описывающих ключ. Возвращается закрытый ключ SK.

Шифрование.

Алгоритм шифрования принимает открытые параметры PK, сообщение M и структуру доступа Λ над множеством атрибутов. Алгоритм шифрует M и производит шифртекст CT, такой, что только пользователь, обладающий набором атрибутов, удовлетворяющих структуре доступа, будет способен расшифровать сообщение. Будем считать, что зашифрованный текст неявно содержит Λ.

Расшифрование.

Алгоритм расшифрования принимает параметры PK, шифртекст CT, который содержит политику доступа Λ, секретный ключ SK, который является секретным ключом для множества атрибутов S. Если набор атрибутов S удовлетворяет структуре доступа Λ, то алгоритм расшифрует шифртекст и вернёт сообщение M.

Делегат.

Алгоритм принимает на вход параметры секретного ключа SK для некоторого набора атрибутов S и набор $\hat{S} \subseteq S$. Возвращает секретный ключ $\widehat{SK}$ набора атрибутов $\hat{S}$. Пусть G_1, G_2 – билинейные группы простого порядка p. Элемент g– образующий элемент группы G_1. Отображение $\varphi: G_1 \times G_1 \to G_2$- билинейное.

Определим коэффициент Лагранжа $\Delta_{(i,s)}$ для $i \in Z_p$ и набора S элементов из Z_p:

$$\Delta_{(i,s)}(x) = \prod_{(j \in S, j \neq i)} \frac{x-j}{i-j} \quad (18)$$

Определим хеш-функцию $H: \{0,1\}^* \to G_1$, которая будет отображать любой атрибут, представленный в двоичном виде, в элемент группы G_1.

Рассмотрим схему подробнее.

Инициализация

$Setup()$: Доверенный центр выбирает билинейную группу G_1 простого порядка p, g – образующий элемент группы G_1. Выбираются два случайных числа $\alpha, \beta \in Z_p$, тогда открытый ключ PK и мастер ключ MK:

$PK = (G_1, g, h, f, \varphi(g,g)^{\alpha})$ (19)

$MK = (\beta, g^{\alpha})$ (20)

Где $h = g^{\beta}, f = g^{(\frac{1}{\beta})}$ (21)

Генерация закрытого ключа пользователя

$KeyGen(S, MK)$: Алгоритм генерации ключа принимает в качестве входных данных набор атрибутов S и возвращает ключ, которые ассоциированы с этим набором атрибутов. Выбирается случайное число $r \in Z_p$, тогда секретный ключ:

$SK = (D, \{D_j, D_j^*\}_{j \in s}$ (22)

Где $D = g^{\frac{\alpha+r}{\beta}}, D_j = g^r H(j)^{r_j}, D_j^* = g^{r_j}$ (23)

Шифрование

$Encrypt(\tau, PK, M)$: Для шифрования выбирается многочлен q_x для каждого узла x в дереве τ. Многочлены выбираются сверху вниз, начиная с корня R следующим образом. Для каждого узла x дерева задаётся многочлен q_x степени $d_x = d_k - 1$, причем $q_x(0) = q_{parant(x)}\big(index(x)\big)$. Начиная с корня R выбирается случайное число $s \in Z_p$ такое, что $q_R(0) = s$. Затем случайно

выбирается d_R значений для того, чтобы полностью определить многочлен q_R. Для других узлов x выбирается d_x значений для того, чтобы полностью определить многочлен q_x.

Пусть Y – множество листьев в τ, тогда шифртекст есть следующий набор:

$$CT = (\tau, \hat{C}, C, \{C_y, C_y^*\}_{y\in\tau} \quad (24)$$

Где $\hat{C} = M\varphi(g,g)^{\alpha s}, C = h^S, C_y^* = H(att(y)^{q_y(0)})$. (25)

Делегат

Delegate(SK, S): Выбираются случайные числа $\hat{r}$ $\widehat{r_k}$ для любых $k \in \hat{S}$, где $\hat{S} \subseteq S$ формируется новый закрытый ключ:

$$\widehat{SK} = (\widehat{D}, \{\widehat{D_k}, \widehat{D_k^*}\}_{k\subseteq\hat{S}}), \textit{(26)}$$

Где $\widehat{D} = Df^{\hat{r}}, \widehat{D_k} = D_k g^{\hat{r}} H(k)^{\widehat{r_k}}, \widehat{D_k^*} = D_k^* g^{\hat{r}}$ (27)

Расшифрование

$Decrypt(CT, SK)$:Определим рекурсивынй алгоритм $DecryptNode(CT, SK, x)$ следующим образом. Если узелx – лист, то $i = att(x)$ и если $i \in S$, то

$$DecryptNode(CT, SK, x) = \frac{\varphi(D_i, C_x)}{\varphi(D_i^*, C_x^*)} = \frac{\varphi(g^r H(i)^{r_i}, h^{q_x(0)})}{\varphi(g^{r_i}, H(i)^{q_x(0)})} = \varphi(g,g)^{rq_x(0)} (28)$$

Если $i \notin S$, то

$$DecryptNode(CT, SK, x) = 1 \ (29)$$

Если x не является листом, то для всех узлов-наследников z узла x рекурсивно вызывается $DecryptNode(CT, SK, z)$, результат сохраняется и обозначается как Fz. Пусть Sx – случайное множество потомков узла z таких, что $F_x \neq \perp$, $|S_x| = k_x$. Если $S_x = \emptyset$, то функция возвратит $\perp$. В противном случае вычисляется следующее значение

$$F_x = \prod_{z \in S_x} F(x)^{\Delta_{i,S_{x^*}}(0)} = \prod_{z \in S_x} (\varphi(g,g)^{rq_x(0)})^{\Delta_{i,S_{x^*}}(0)} \quad (30)$$

$$\prod_{z \in S_x} (\varphi(g,g)^{rq_{patent(x)}(index(z))})^{\Delta_{i,S_{x^*}}(0)} \quad (31)$$

$$\prod_{z \in S_x} (\varphi(g,g)^{rq_x(i)})^{\Delta_{i,S_{x^*}}(0)} = \varphi(g,g)^{rq_x(0)} \quad (32)$$

где $i = index(z)$, $S_{x^*} = \{index(z) : z \in S_x\}$

Расшифрование начинается с вызова функции $DecryptNode$ для корня R дерева τ, если набор атрибутов удовлетворяет дереву τ, то вычисляется $A = DecryptNode(CT, SK, r) = \varphi(\varphi(g,g)^{rq_R(0)} = \varphi(g,g)^{rs}$ (33)

тогда исходное сообщение

$$M = \frac{\hat{C}}{\varphi(C,D)/A} = \frac{\hat{C}}{\varphi(h^*, g^{\frac{\alpha+r}{\beta}})/\varphi(g,g)^{rs}} \quad (34)$$

В данной работе были рассмотрены криптосистемы на основе атрибутов. Описаны задачи, которые могут быть решены с помощью этой технологии. Был проведен анализ криптосистем на основе идентификаторов и криптосистем на основе атрибутов. На наш взгляд, схема шифрования на основе атрибутов идеальна для закрытых групп пользователей, таких как руководители многонациональной компании или филиалов крупных банков, так как штаб-

квартира корпорации может служить центром генератора ключей, которому доверяет каждый пользователь.

Библиографический список:

1. Gentry C. Certificate-based encryption and the certificate revocation problem. International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, 2003. C. 272–293
2. Adi Shamir, Identity-Based Cryptosystems and Signature Schemes. Advances in Cryptology: Proceedings of CRYPTO 84, Lecture Notes in Computer Science, 7:47--53, 1984
3. Amit Sahai and Brent Waters, Fuzzy Identity-Based Encryption Cryptology ePrint Archive, Report 2004/086 (2004)
4. Cheng-Chi Lee, Pei-Shan Chung, and Min-Shiang Hwang, A Survey on Attribute-based Encryption Schemes of Access Control in Cloud Environments, International Journal of Network Security, Vol.15, No.4, PP.231-240, July 2013
5. Vipul Goyal, Omkant Pandey, Amit Sahai and Brent Waters, Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data ACM CCS (2006)

Оригинальность 83%