Секция 4

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

УДК 004.75 DOI 10.17223/2226308X/9/30

ПРОТОКОЛ БЕЗОТКАЗНОЙ ЛУКОВОЙ МАРШРУТИЗАЦИИ С ПОДТВЕРЖДЕНИЕМ ВРЕМЕНИ СОЗДАНИЯ СООБЩЕНИЯ

Н. И. Анисеня

Рассматривается задача маршрутизации сообщений в конкурентной среде децентрализованной сети на примере проведения соревнований CTF, основанных на решении заданий. К протоколу маршрутизации сообщений предъявляется дополнительное требование — подтверждение времени создания сообщения. Предложено улучшение протокола безотказной луковой маршрутизации, позволяющее участникам сети получать подтверждения времени создания передаваемых сообщений, описан улучшенный протокол и его возможная модификация.

Ключевые слова: *распределённые протоколы, защищённые вычисления, отказоустойчивые системы.*

В [1] автором предложен распределённый протокол проведения соревнований CTF, основанных на решении заданий. Описанный протокол обладает серьёзным недостатком, а именно — полагается на рассылку меток времени добросовестными участниками. Недобросовестный участник, который не рассылает свои разрешения, все ещё имеет доступ к сети и может получать сообщения, если участвует в передаче сообщений других команд.

Цель улучшения протокола безотказной луковой маршрутизации — сделать невозможным участие в соревновании команд, которые не ставят метки времени либо ставят некорректные метки под ответами других участников.

Будем полагать, что каждый участник сети постоянно ожидает получения сообщений. Сформулируем требования к улучшенному протоколу:

1) каждый участник сети должен иметь возможность получать сообщения только в том случае, если участвует в качестве посредника при передаче сообщений других участников;
2) каждый участник сети должен иметь возможность получать сообщения только в том случае, если устанавливает корректные метки времени на все передаваемые им сообщения;
3) в результате работы протокола отправитель сообщения должен иметь временные метки от различных участников сети, подтверждающие время отправки исходного сообщения;
4) никто из участников, кроме отправителя и получателя, не должен иметь возможность прочитать или изменить передаваемое сообщение;
5) никто из участников не должен иметь возможность подделать чужую метку времени под передаваемым сообщением;

6) протокол должен делать крайне сложной и маловероятной атаку сговора с целью установки некорректных меток времени под сообщением;
7) протокол должен допускать особый режим работы, если нет гарантий, что получатель сообщения в данный момент подключён к сети, но при этом необходимо здесь и сейчас зафиксировать время создания сообщения. В свою очередь, получатель должен иметь возможность убедиться в подлинности временной метки под сообщением, которое он получит, когда восстановит доступ в сеть.

Требования 1 и 2 необходимы для того, чтобы участники распределённой сети были заинтересованы в поддержании её работы. Требования 4 и 5 необходимы для защиты передаваемых сообщений и устанавливаемых под ними меток времени. Требование 6 служит для защиты от атаки сговора участников распределённой сети. Требование 7 необходимо, если данный протокол используется для проведения соревнований CTF, как в [1]. В этом случае необходимо уменьшить зависимость протокола от активного участия команды организаторов.

Пусть E_A — шифрование на открытом ключе участника с идентификатором A; S_A — подписание на закрытом ключе участника с идентификатором A; h — некоторая криптографическая хеш-функция. В случае, если y является конкатенацией байтового представления параметров $x_1, \ldots, x_t$, будем допускать следующую запись для функций E_A, S_A, h:

$$\begin{aligned} E_A(y) &= E_A(x_1, \ldots, x_t), \\ S_A(y) &= S_A(x_1, \ldots, x_t), \\ h(y) &= h(x_1, \ldots, x_t). \end{aligned}$$

Будем проводить сравнение значений времени с учётом временного окна w. Обозначим множество идентификаторов участников как U, $|U| = n$, где идентификаторы $id \in U$ — попарно различные числа. Функцию $G(y)$ определим как в [1]:

$$G(y) : \mathbb{N} \to U', \quad U' = \{u : u \subset U, |u| = t < n\}.$$

Введём функцию $H(y)$ следующим образом:

$$\begin{gathered} H(y) = [r_0, (id_1, r_1), \ldots, (id_t, r_t)], \\ id_i \in G(y),\ id_i < id_{i+1},\ i = 1, \ldots, t-1, \\ r_0 = h(y),\ r_i = h(y, id_i),\ i = 1, \ldots, t. \end{gathered}$$

Для удобства обозначим так же векторы идентификаторов участников и соответствующих им чисел:

$$M = (id_1, \ldots, id_t),\ R = (r_0, r_1, \ldots, r_t).$$

Будем говорить, что участник отключен от сети, если он не участвует в передаче сообщений либо искажает передаваемые им сообщения. Если все участники с идентификаторами $id_1, \ldots, id_t$ выступают в роли посредников при передаче сообщения, будем обозначать эти идентификаторы $M_1, \ldots, M_t$. Соответственно вектор M в этом случае запишется следующим образом: $M = (M_1, \ldots, M_t)$.

Пусть Алиса хочет послать сообщение m Бобу. При этом Алиса заинтересована в следующем:

1) она хочет убедиться в том, что все посредники, участвовавшие в передаче сообщения m, и получатель Боб честно выставляют метку времени (не завышают и не занижают время), когда выступают в роли посредника;

2) она хочет получить подтверждение времени создания сообщения m от других участников и предоставить его Бобу.

Боб, в свою очередь, хочет быть уверенным, что Алиса его не обманывает и указанное время создания сообщения m верно с точностью до некоторого временного окна w. Будем предполагать, что все участники сети постоянно ожидают сообщений и не могут предсказать время, в которое они должны им прийти.

Описание протокола:

1) Алиса считает $H(m) = [r_0, (M_1, r_1), \ldots, (M_t, r_t)]$.
2) Алиса конструирует «луковицу», шифруя сообщение m следующим образом:

$$E_B(E_{M_1}(\ldots E_{M_k}(E_A E_B(m), r_k), \ldots, r_1), r_0).$$

3) Алиса отправляет Бобу

$$E_B(E_{M_1}(\ldots E_{M_k}(E_A E_B(m), r_k), \ldots, r_1), r_0), \quad S_A(m, t_A), t_A,$$

где t_A — текущее время Алисы.

4) Боб расшифровывает внешний слой луковицы: получает r_0 и сообщение для первого посредника M_1. Боб отправляет посреднику M_1

$$E_{M_1}(\ldots E_{M_k}(E_A E_B(m), r_k), \ldots, r_1), \quad S_A(m, t_A), t_A, S_B(r_0, t_B), t_B.$$

5) Для $i \in \{1, \ldots, k-1\}$ i-й посредник отсылает $(i+1)$-му следующее сообщение:

$$E_{M_i}(E_{M_{i+1}}(\ldots E_{M_k}(E_A E_B(m), r_k), \ldots, r_{i+1}), r_i),$$
$$S_A(m, t_A), t_A, S_B(r_0, t_B), t_B, S_{M_1}(r_1, t_1), t_1, \ldots, S_{M_i}(r_i, t_i), t_i.$$

6) Последний посредник M_k отправляет Алисе

$$E_A E_B(m), \quad S_A(m, t_A), t_A, S_B(r_0, t_B), t_B, S_{M_1}(r_1, t_1), t_1, \ldots, S_{M_k}(r_k, t_k), t_k.$$

7) Алиса проверяет, что все метки времени $S_B(r_0, t_B)$, $S_{M_i}(r_i, t_i)$, t_i, $i = 1, \ldots, k-1$, имеют корректную подпись и указывают на адекватное время. Если проверки пройдены, Алиса посылает Бобу

$$E_B(m), \quad S_A(m, t_A), t_A, S_B(r_0, t_B), t_B, S_{M_1}(r_1, t_1), t_1, \ldots, S_{M_k}(r_k, t_k), t_k.$$

Боб проводит проверку следующим образом.

1) Посчитать $H(m) = [r_0, (M_1, r_1), \ldots, (M_k, r_k)]$.
2) Проверить подписи временных меток $S_A, S_B, S_{M_i}, i = 1, \ldots, k$.
3) Проверить, что для значений времени в метках времени верны следующие равенства:

$$t_{M_i} =_w t_{M_{i+1}}, \ i \in \{1, \ldots, k-1\}, \quad t_{M_1} =_w t_B, \quad t_B =_w t_A.$$

4) Если проверки пройдены, то считать, что сообщение m создано в момент времени t_A.

В случае, если Боб в данный момент не подключен к сети, а Алиса хочет засвидетельствовать факт создания сообщения m немедленно, можно внести следующие модификации в протокол. На шаге 2 внешний слой шифрования для Боба не включается.

Последний шаг 7 протокола не зависит от времени, поэтому может быть выполнен, когда Боб снова подключится к сети. Проверка выполняется аналогично с учётом того, что временная метка $S_B(r_0, t_B), t_B$ отсутствует.

Некоторые комментарии к протоколу. Функция $H(m)$, аналогично функции $G(m)$, имеет параметры n, t, k. В случе, если хотя бы один из первых k посредников из M отключен от сети, получение меток времени от этой цепочки посредников невозможно. Для того чтобы получить метку времени, необходимо выбрать k подключенных участников и сформировать луковицу согласно порядку в M. Таким образом, $H(m)$ также имеет запас в $(t-k)$ участников.

Функция $G(m)$ в составе функции $H(m)$ нужна для того, чтобы случайным образом выбирать участников для выставления метки времени под сообщением m, тем самым сильно затрудняя возможность их предварительного сговора с целью установки ложной метки времени.

Вектор R, получаемый функцией $H(m)$, является вектором чисел, которые невозможно предсказать, не зная сообщения m. Это нужно для того, чтобы посредники имели возможность поставить временную метку на $r_i = h(m, id_i)$, не зная самого сообщения m. Числа r_i также используются в качестве *Nonce*, чтобы следующие посредники не могли определить предыдущих посредников, которые уже поставили свою метку времени на сообщение.

Рассмотрим пример. Имеется множество участников $U = \{A, B, M_1, M_2\}$, $|U| = n = 4$, $t = k = 2$. Они подключены к сети, как показано на рис. 1.

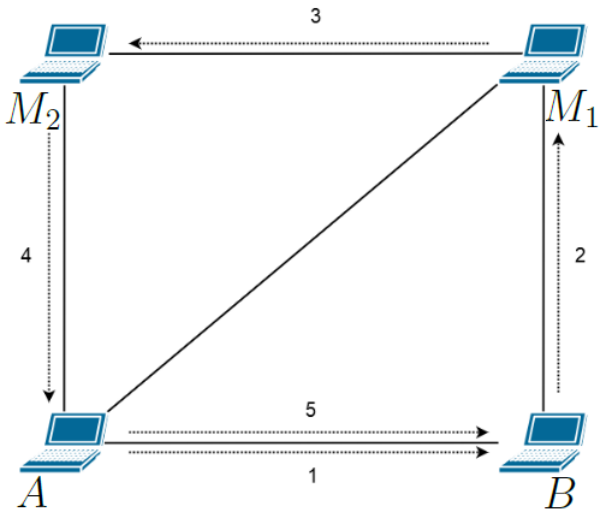

Рис. 1. Пример работы протокола безотказной луковой маршрутизации с подтверждением времени создания сообщения

Пусть участник A хочет послать сообщение m участнику B. Для этого ему необходимо вычислить $H(m) = [r_0, (M_1, r_1), (M_2, r_2)]$. Имеем следующую луковицу $E_B(E_{M_1}(E_{M_2}(E_A E_B(m), r_2), r_1), r_0)$. Подробно рассмотрим процесс передачи луковицы и выставления меток времени:

1) $A \to B$: $E_B(E_{M_1}(E_{M_2}(E_A E_B(m), r_2), r_1), r_0)$, $S_A(m, t_A), t_A$.
2) $B \to M_1$: $E_{M_1}(E_{M_2}(E_A E_B(m), r_2), r_1)$, $S_A(m, t_A), t_A, S_B(r_0, t_b), t_B$.

3) $M_1 \to M_2$: $E_{M_2}(E_A E_B(m), r_2)$,
$S_A(m, t_A), t_A, S_B(r_0, t_b), t_B, S_{M_1}(r_1, t_{M_1}), t_{M_1}$.

4) $M_2 \to A$: $E_A E_B(m)$,
$S_A(m, t_A), t_A, S_B(r_0, t_b), t_B, S_{M_1}(r_1, t_{M_1}), t_{M_1}, S_{M_2}(r_2, t_{M_2}), t_{M_2}$.

5) Проверка: $t_{M_2} =_w t_{M_1}$, $t_{M_1} =_w t_B$, $t_B =_w t_A$. Если проверки пройдены:
$A \to B$: $E_B(m)$,
$S_A(m, t_A), t_A, S_B(r_0, t_b), t_B, S_{M_1}(r_1, t_{M_1}), t_{M_1}, S_{M_2}(r_2, t_{M_2}), t_{M_2}$.

Описанный протокол может использоваться как протокол транспортного уровня в распределённой сети соревнования CTF следующим образом. Команда организаторов, согласно протоколу, осуществляет новостную рассылку во время соревнования. Если участник хочет получать новостную рассылку от организаторов, ему необходимо выступать в качестве посредника при передаче сообщений других участников, выставляя на них корректную метку времени. В противном случае участник не сможет получать новостную рассылку, что равносильно отключению от соревнования и техническому поражению.

Каждый участник при получении ответа на задание должен доказать этот факт, отправив сообщение организаторам. Засвидетельствовать факт получения ответа в текущий момент времени можно с помощью модифицированного протокола, т. е. предполагая, что команда организаторов в данный момент может быть недоступна. Команды-соперники, выбранные в качестве посредников с помощью функции $H(m)$, подтвердят факт создания сообщения в текущий момент времени. Итоговое сообщение вместе с метками времени может быть отправлено организаторам позже. По окончании соревнования организаторы смогут сформировать таблицу результатов с учётом времени получения ответов.

ЛИТЕРАТУРА

1. *Анисеня Н. И.* Разработка безопасного протокола распределённой системы проведения соревнований CTF // Прикладная дискретная математика. 2015. № 2(28). С. 59–70.

УДК 519.7 DOI 10.17223/2226308X/9/31

О ДИСКРЕТНО-АВТОМАТНЫХ МОДЕЛЯХ АТАК В КОМПЬЮТЕРНЫХ СЕТЯХ[1]

Д. Е. Горбатенко, С. Е. Кочемазов, А. А. Семёнов

Предлагается новая модель развития атак в компьютерных сетях. Основу модели составляет дискретный автомат синхронного действия, задаваемый графом сети. Рассматриваются переходы между состояниями данного автомата, совершаемые в дискретные моменты времени. Вершинам графа (интерпретирующим хосты сети) в каждый момент времени приписываются двоичные векторы, называемые состояниями хостов. В каждый следующий момент все состояния хостов синхронно пересчитываются по фиксированным правилам. В рамках предложенной модели проанализировано развитие некоторых известных типов атак. Изучены возможности атак с нескольких хостов, а также описана техника предотвращения атак посредством решения комбинаторной задачи расстановки патчей. В вычислительных экспериментах рассматривались сети, сгенерированные случайным образом. Перечисленные выше задачи для этих сетей решались за счёт их сведения к проблеме булевой выполнимости.

[1]Работа выполнена при частичной поддержке РФФИ, проекты № 14-07-00403а и 15-07-07891а.