РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 684 495**(13) **C1**

(51) МПК
H04L 12/823 (2013.01)
H04L 29/06 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК
***H04L 12/6418** (2019.02); **H04L 47/32** (2019.02); **H04L 69/22** (2019.02); **H04L 69/324** (2019.02); **H04L 29/06** (2019.02)*

(21) (22) Заявка: **2018113078, 11.04.2018**

(24) Дата начала отсчета срока действия патента:
11.04.2018

Дата регистрации:
09.04.2019

Приоритет(ы):
(22) Дата подачи заявки: **11.04.2018**

(45) Опубликовано: **09.04.2019** Бюл. № **10**

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр. 1, Открытое акционерное общество "Информационные технологии и коммуникационные системы"

(72) Автор(ы):
Паршин Илья Анатольевич (RU),
Тычина Леонид Анатольевич (RU)

(73) Патентообладатель(и):
Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)

(56) Список документов, цитированных в отчете о поиске: **RU 2535172 C2, 10.12.2014. US 2007/0083923 A1, 12.04.2007. US 2010/0165839 A1, 01.07.2010. US 9137139 B2, 15.09.2015.**

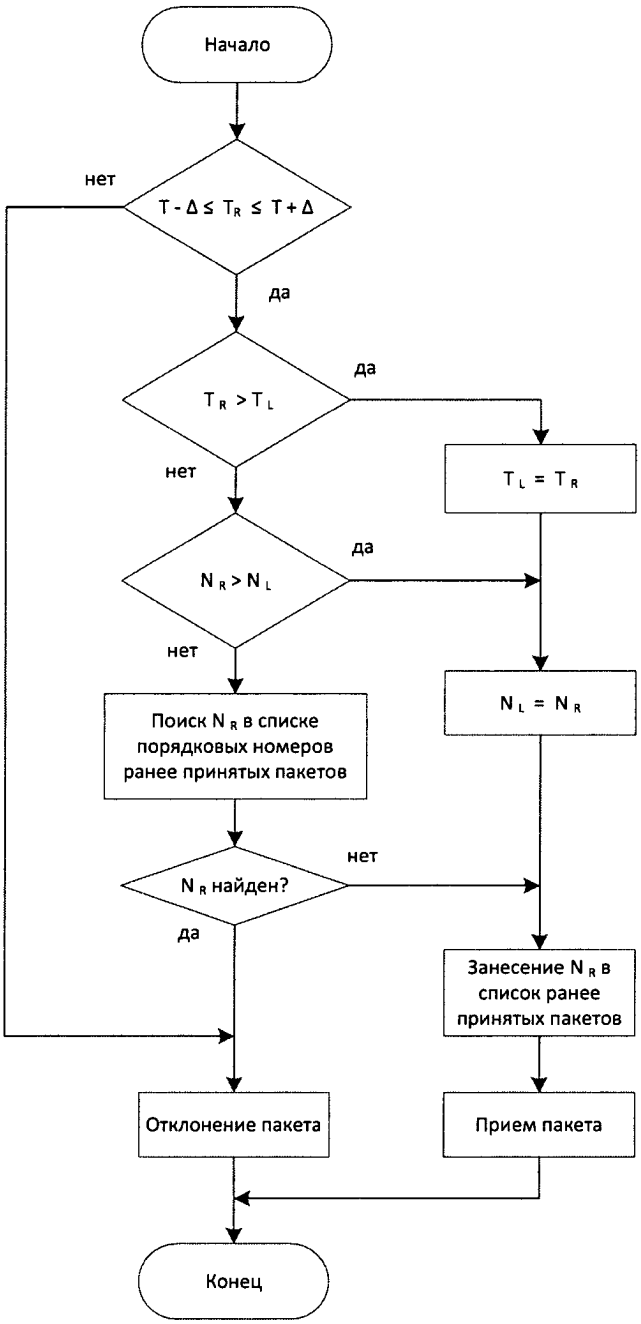
(54) Способ предотвращения повторного использования пакетов цифровых данных в сетевой системе передачи данных

(57) Реферат:

Изобретение относится к обеспечению безопасности в сетях передачи данных. Технический результат – предотвращение повторного приема пакетов цифровых данных в сетевой системе передачи данных. Способ предотвращения повторного использования пакетов цифровых данных в сетевой системе передачи данных, в котором получают в выбранном шлюзе полезные данные для каждого отправляемого пакета, формируют метаданные для каждого отправляемого пакета, причем метаданные включают номер пакета, время отправки пакета данных, данные для проверки целостности метаданных, отправляют пакет из выбранного шлюза через сеть передачи данных, устанавливают на компьютере, принимающем сообщения, допустимую величину промежутка времени рассогласования, формируют в памяти компьютера, принимающего сообщения, области для хранения времени отправки и номера последнего принятого пакета, списка номеров ранее принятых пакетов данных от каждого отправителя, принимают пакет, включающий полезные данные и метаданные, проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных, проводят проверку на повтор принятого пакета, принимают пакет, включающий полезные данные и метаданные, проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных, проводят проверку на повтор принятого пакета, при этом если время отправки принятого пакета выходит за пределы промежутка времени рассогласования, то отклоняют пакет, если время отправки пакета находится в пределах промежутка времени рассогласования, то принимают пакет. 3 ил., 1 табл.

RU 2684495 C1

RU 2684495 C1



Фиг. 2

RU 2684495 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) ABSTRACT OF INVENTION

(19) RU (11) 2 684 495 (13) C1

(51) Int. Cl.
H04L 12/823 (2013.01)
H04L 29/06 (2006.01)

(52) CPC
H04L 12/6418 (2019.02); H04L 47/32 (2019.02); H04L 69/22 (2019.02); H04L 69/324 (2019.02); H04L 29/06 (2019.02)

(21) (22) Application: **2018113078, 11.04.2018**

(24) Effective date for property rights:
11.04.2018

Registration date:
09.04.2019

Priority:
(22) Date of filing: **11.04.2018**

(45) Date of publication: **09.04.2019** Bull. № **10**

Mail address:
127287, Moskva, Staryj Petrovsko-Razumovskij pr-d, 1/23, str. 1, Otkrytoe aktsionernoe obshchestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy"

(72) Inventor(s):
Parshin Ilya Anatolevich (RU),
Tychina Leonid Anatolevich (RU)

(73) Proprietor(s):
Otkrytoe aktsionernoe obshchestvo "Informatsionnye tekhnologii i kommunikatsionnye sistemy" (RU)

RU 2684495 C1

(54) METHOD OF PREVENTING REUSE OF DIGITAL DATA PACKETS IN A NETWORK DATA TRANSMISSION SYSTEM

(57) Abstract:

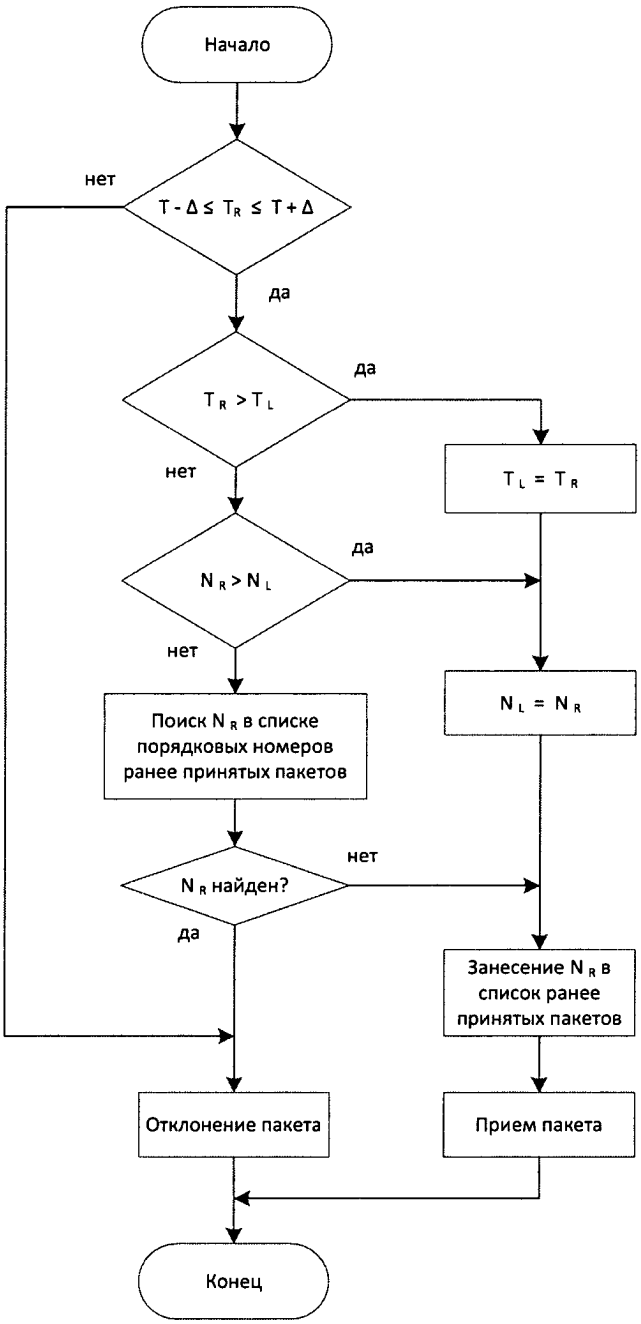
FIELD: information technology.

SUBSTANCE: invention relates to providing security in data transmission networks. Method for preventing reuse of digital data packets in a network data transmission system, in which obtaining, in a selected gateway, payload data for each sent packet, generating metadata for each sent packet, metadata including a packet number, a data packet sending time, metadata integrity checking data, sending a packet from the selected gateway through the data transmission network, establishing, on a computer receiving messages, the mismatch time tolerance value, storing the sending time and number of the last received packet, the list of numbers of previously received data packets from each sender in the memory of the receiving computer, receiving a packet which includes payload data and metadata, validating metadata integrity of the received packet, using the metadata integrity checking data, checking for re-reception of the received packet, receiving a packet which includes payload data and metadata, validating metadata integrity of the received packet, using the metadata integrity checking data, checking for re-reception of the received packet, wherein if the time of sending the received packet exceeds the time interval of the mismatch, then rejecting the packet, if the time for sending the packet is within the time interval of the discrepancy, then the packet is received.

EFFECT: preventing repeated reception of digital data packets in a network data transmission system.

1 cl, 3 dwg, 1 tbl

RU 2684495 C1



Фиг. 2

RU 2684495 C1

Область техники, к которой относится изобретение

Предполагаемое изобретение относится к способам обеспечения безопасности в сетях передачи данных и, в частности, к способам предотвращения повторного использования пакетов цифровых данных при передаче в сетях передачи данных.

Уровень техники

В сети передачи данных с использованием цифровых пакетов данных, например, по протоколу IP, могут использоваться различные протоколы для обеспечения безопасности IP сетей, например протокол IPsec [RFC 4302 - Идентификационный заголовок IP, 2005, материал по адресу http://rfc2.ru/4302.rfc; RFC 4303 - Инкапсуляция защищенных данных IP (ESP), материал по адресу http://rfc2.ru/4303.rfc].

В протоколе IPsec предусмотрен способ предотвращения повторного использования (Anti-replay) пакетов цифровых данных в процессе установленного соединения между компьютерами в сети. Этот способ включает использование автоматической нумерации передаваемых пакетов после установления соединения, включения номера передаваемого пакета в стандартный заголовок пакета и учета номеров принятых пакетов.

При приеме очередного пакета в компьютере, принимающем сообщения, происходит процедура поиска номера принятого пакета в списке ранее принятых пакетов, и, если пакет был ранее принят, то он отклоняется. Если же пакет не был ранее принят, то его номер вносится в список, и пакет принимается и обрабатывается.

При приеме зашифрованных пакетов применяются предварительно операции расшифрования и контроля целостности.

Недостатками способа являются:

1) необходимость протокола согласования счетчиков и, как следствие, необходимость двухсторонней фиксации факта создания соединения;

2) необходимость синхронизации обращений к счетчику порядковых номеров на стороне отправителя при наличии нескольких потоков исполнения (что замедляет обработку передаваемой информации);

3) появление коллизий номеров пакетов при наличии нескольких отправителей-шлюзов.

Известен также способ предотвращения повторного использования пакетов цифровых данных в сетевой системе передачи данных (патент США №9137139, приоритет от 18.12.2009 г.), причем система включает

- компьютеры или шлюзы безопасности, отправляющие сообщения в виде последовательности пакетов цифровых данных через сеть передачи данных, и
- компьютер или шлюз безопасности, принимающий сообщения и связанные с другими компьютерами и шлюзами безопасности через сеть передачи данных;

способ, заключающийся в том, что

- формируют полезные данные для каждого отправляемого пакета;
- формируют метаданные для каждого отправляемого пакета, причем метаданные включают
  - время отправки пакета данных (временную метку, pseudo-time stamp),
  - номер отправителя и порядковый номер пакета (порядковые номера ведутся каждым отправителем независимо) или номер отправителя и порядковый номер пакета, в роли которого выступает временная метка;
- формируют пакет, включающий полезные данные и метаданные;
- шифруют пакет вместе с метаданными;
- отправляют пакет через сеть передачи данных;
- устанавливают на компьютере, принимающем сообщения, допустимую величину

промежутка времени рассогласования;

- формируют в памяти компьютера области для хранения
  - времени отправки последнего принятого пакета;
  - списка порядковых номеров последних принятых пакетов - по одному элементу для каждого отправителя;
  - списка списков порядковых номеров ранее принятых пакетов - по одному списку на каждого отправителя;
- принимают пакет, включающий полезные данные и метаданные;
- расшифровывают пакет вместе с метаданными, включая время отправки пакета данных;
- проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных;
- проводят проверку на повтор принятого пакета, выполняя следующие действия
  - если время отправки принятого пакета выходит за пределы промежутка времени рассогласования, то отклоняют пакет;
  - если время отправки пакета находится в пределах промежутка времени рассогласования и номер пакета не является меткой времени, то
    - если время отправки принятого пакета больше времени отправки последнего принятого пакета, то заменяют время отправки последнего принятого пакета на время отправки принятого пакета;
    - если порядковый номер принятого пакета меньше любого из номеров ранее принятых от соответствующего отправителя пакетов, то отклоняют пакет;
    - если пакет с порядковым номером, соответствующим порядковому номеру принятого пакета, уже принимался от соответствующего отправителя, то отклоняют пакет;
    - если порядковый номер принятого пакета больше меньшего из номеров ранее принятых от соответствующего отправителя пакетов и меньше наибольшего, то помечают, что, от соответствующего отправителя получен пакет с данным порядковым номером;
    - если порядковый номер принятого пакета больше любого из номеров ранее принятых от соответствующего отправителя пакетов, то порядковый номер принятого пакета запоминается как номер последнего принятого от соответствующего отправителя пакета и заносится в список номеров принятых пакетов, а сам список укорачивается;
  - принимают пакет для последующей обработки.

Для формирования времени отправки пакета данных в компьютере, отправляющем сообщения, могут использоваться встроенные (локальные) часы или значения, полученные от внешнего сервера.

Для контроля времени отправки пакета данных в компьютере, принимающем сообщения, предусмотрено использование встроенных (локальных) часов.

Описанный способ принимается за прототип.

Однако, известный способ не позволяет вести обработку отправляемых пакетов на максимальной для данного оборудования скорости, кроме того, компьютер или шлюз безопасности, получающий пакеты, должен иметь сведения о количестве отправляющих пакеты шлюзов безопасности и вести для каждого из них отдельный учет принятых пакетов, что является недостатками известного способа.

Раскрытие изобретения

Техническим результатом является предотвращение (устранение возможности) повторного приема пакетов.

Дополнительным техническим результатом является:

1) отсутствие необходимости двухсторонней фиксации факта создания соединения;

2) отсутствие необходимости передавать получателю пакетов информации о количестве реальных отправителей;

3) отсутствие необходимости ведения нескольких счетчиков для одного потока информации на стороне принимающего компьютера;

4) обработка отправляемых пакетов на максимальной для отправляющего оборудования скорости.

Для этого предлагается способ, реализуемый с помощью системы, включающей

- несколько шлюзов безопасности, выполненных с возможностью формировать пакеты и имеющие нескольких процессорных блоков, причем каждый процессорный блок имеет отдельный счетчик отправленных пакетов;
- компьютер, принимающие сообщения и связанный с другими компьютерами через сеть передачи данных;

способ, заключающийся в том, что

- получают в выбранном шлюзе полезные данные для каждого отправляемого пакета;
- формируют метаданные для каждого отправляемого пакета, причем метаданные включают
  - номер пакета, вычисляемый по формуле

$$NU = CP * 2^{NS + NG} + NT * 2^{NG} + NA, \qquad (1)$$

где CP - значение счетчика отправляемых пакетов, независимо используемого данным процессором выбранного шлюза безопасности

NS - количество двоичных разрядов, требуемое для хранения номера процессора в выбранном шлюзе безопасности,

NG - количество двоичных разрядов, требуемое для хранения номера шлюза безопасности,

NT - номер процессора выбранного шлюза безопасности, формирующего данный пакет,

NA - номер шлюза безопасности, формирующего данный пакет, причем

$NS=[\log_2 NV]$,

где NV - общее количество процессоров в данном шлюзе безопасности, осуществляющем защиту трафика,

[x] - операция, возвращающая наименьшее целое число, большее или равное указанному числу x;

$NG=[\log_2 NB]$,

где NB - общее количество шлюзов безопасности, осуществляющих защиту трафика;

  - время отправки пакета данных;
  - данные для проверки целостности метаданных;
- отправляют пакет из выбранного шлюза через сеть передачи данных;
- устанавливают на компьютере, принимающем сообщения, допустимую величину промежутка времени рассогласования;
- формируют в памяти компьютера принимающего сообщения, области для хранения следующих данных от каждого отправителя:
  - времени отправки последнего принятого пакета,
  - номера последнего принятого пакета,
  - списка номеров ранее принятых пакетов;

- принимают пакет, включающий полезные данные и метаданные;
- проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных;
- проводят проверку на повтор принятого пакета, выполняя следующие действия с использованием хранимых в памяти данных для данного отправителя:
  - времени отправки последнего принятого пакета,
  - номера последнего принятого пакета,
  - списка номеров ранее принятых пакетов;
- принимают пакет, включающий полезные данные и метаданные;
- проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных;
- проводят проверку на повтор принятого пакета, выполняя следующие действия
  - если время отправки принятого пакета выходит за пределы промежутка времени рассогласования, то отклоняют пакет;
  - если время отправки пакета находится в пределах промежутка времени рассогласования, то
    - если время отправки принятого пакета больше времени отправки последнего принятого пакета, то
      - заменяют время отправки последнего принятого пакета на время отправки принятого пакета;
      - заменяют номер последнего принятого пакета на номер принятого пакета;
      - заносят номер принятого пакета в список номеров ранее принятых пакетов;
      - принимают пакет;
    - если время отправки принятого пакета не превышает времени отправки последнего принятого пакета, то
      - если номер последнего принятого пакета меньше номера принятого пакета, то
        - ✓ заменяют номер последнего принятого пакета на номер принятого пакета;
        - ✓ заносят номер принятого пакета в список номеров ранее принятых пакетов;
        - ✓ принимают пакет;
      - если номер последнего принятого пакета больше или равен номеру принятого пакета, то
        - ✓ проводят поиск номера принятого пакета в списке номеров ранее принятых пакетов;
        - ✓ если номер принятого пакета найден в списке номеров ранее принятых пакетов, то отклоняют принятый пакет;
        - ✓ если номер принятого пакета не найден в списке номеров ранее принятых пакетов, то
          - ➢ заносят номер принятого пакета в список номеров ранее принятых пакетов;
          - ➢ принимают принятый пакет.

В отличие от прототипа, где для изменения счетчика каждый процессор должен дождаться своей очереди (так как в случае нерегулируемого изменения счетчика в сеть будут отправлены пакеты с одинаковыми номерами), что приводит к паузам в обработке пакетов, в предлагаемом решении каждый процессор ведет свой собственный счетчик, не задерживая другие процессоры, что устраняет паузы, вызванные ожиданием своей очереди изменить значение счетчика, и повышает скорость обработки пакетов, тем самым обеспечивая максимальную скорость обработки пакетов.

Структура сформированного пакета приведена на фиг. 1.

Процесс проверки на повтор принятого пакета поясняется схемой, приведенной на фиг. 2.

Использование метки времени для синхронизации номеров пакетов позволяет не фиксировать начальный номер, а считать первым номером пакета таковой, полученный после получения большей метки времени.

Дополнительно, в отличие от прототипа, где получатель должен вести несколько счетчиков номеров пакетов для каждого из потоков данных, маршрутизируемых через несколько независимо генерирующих номера пакетов шлюзов безопасности, для того, чтобы избежать коллизий номеров пакетов (и последующего отбрасывания пакетов с совпадающими номерами), получателю нет необходимости вести несколько счетчиков на один поток данных, так как номер шлюза участвует, согласно формуле (1), в формировании номера пакета, да и количество шлюзов безопасности, процессоров может меняться без нотификации получателя.

Необходимо отметить, что предложенный способ работоспособен и при наличии единственного шлюза безопасности, отправляющего пакеты, в этом случае, согласно (1), номер шлюза безопасности не участвует в формировании номера пакета, но процессоры этого шлюза продолжают формировать номера отправляемых пакетов без уменьшения производительности.

Краткое описание чертежей

На фиг. 1 показана структура пакета с номером NU, включающий номер подготовившего его шлюза безопасности и процессора.

На фиг. 2 показана схема, поясняющая процесс проверки на повтор принятого пакета.

В схеме использованы следующие обозначения:

T - текущее время;

Δ - величина промежутка времени рассогласования;

T_R - время отправки принятого пакета;

N_R - номер принятого пакета;

T_L - время последнего принятого пакета;

N_L - номер последнего принятого пакета.

На фиг. 3 показана структура пакета, сформированного третьим процессором второго шлюза.

Осуществление изобретения

Рассмотрим пример реализации предложенного способа в сети для компьютерной системы, включающей шлюзы безопасности, отправляющие сообщения в виде последовательности пакетов цифровых данных через сеть передачи данных, и компьютеры, принимающие сообщения и связанные со шлюзами безопасности через сеть передачи данных.

В качестве программного обеспечения (ПО), обеспечивающего выполнение действий предложенного способа, могут быть использованы специально разработанные (доработанные) программы или функции в составе стандартного сетевого ПО.

Шлюзы-компьютеры в сети могут работать под управлением операционной системы (ОС) общего назначения, например, Microsoft Windows 10, и должны иметь несколько процессоров (процессорных блоков), каждый из которых получает свой номер. Например, в случае наличия в сети двух шлюзов, каждый из которых включает по восемь процессоров, номер каждого пакета, сформированного третьим процессором

второго шлюза, показан на фиг. 3, при этом CP - значение счетчика отправляемых пакетов, независимо используемого третьим процессором второго шлюза безопасности.

Обмен между отправляющим пакеты шлюзом и получателем может быть организован с помощью UDP протокола, где формат тела UDP дейтаграммы имеет следующий вид (табл. 1).

Таблица 1

← 8 байт →	← 8 байт →	← 8 байт →	← 0 и более байт →
Номер пакета	Время отправки	Имитовставка	Полезные данные

В дейтаграмме используются следующие параметры.

Номер пакета - 64 разрядное число, формируемое процессорами в соответствии с формулой 1 и фиг. 1.

Время отправки - время, закодированное в 64 разрядное число, например Unix время.

Имитовставка - данные для проверки целостности, например, в соответствии с ГОСТ 28147-89. Контроль целостности может осуществляться как для всего пакет (с полем «Имитовставка» равным 0), так и только для метаданных (номер, время отправки).

Полезные данные - это данные, для которых применяется механизм предотвращения повторов.

Для непосредственного использования предложенного способа на компьютеры в сети загружают соответствующие программные модули с предварительно заданным и введенным значением A (величина промежутка времени рассогласования), общим количеством и номерами шлюзов, также программные модули каждого шлюза выделяют области памяти для хранения счетчика номеров пакетов, отправляемых данным шлюзом (далее - «счетчик шлюза»), и счетчиков номеров пакетов, отправляемых каждым из процессоров данного шлюза (далее - «счетчик процессора»).

Величина промежутка времени рассогласования определяется, исходя из желаемого времени восстановления связи между отправляющим пакеты шлюзом и получателем, и допустимой разности показания локальных часов на отправляющем пакеты шлюзе и получателе. Для большинства случаев приемлемым будет Δ=5 минут.

Программный модуль инициализирует область памяти для счетчика номеров пакетов на отправку. Для каждого отправителя, от которого будут приниматься пакеты, маршрутизируемые через шлюзы безопасности, подготавливается область памяти для хранения времени отправки последнего принятого пакета, номера последнего принятого пакета, списка номеров ранее принятых пакетов. Список номеров ранее принятых пакетов можно организовать в виде битового массива, например, так, как это описано в описании протокола IPsec (rfc 4303, раздел A2).

Затем каждый процессор шлюза формирует пакеты, каждый из которых включает: полезные данные, номер пакета, время отправки, имитовставку. В поле «Номер пакета» сохраняется модифицированное в соответствии с (1) значение счетчика номеров пакетов на отправку, затем значение счетчика увеличивается на единицу. Поле «Время отправки» содержит значение времени на момент отправки. Поле «Имитовставка» содержит значение имитовставки, рассчитанное по ГОСТ 28147-89 для метаданных (номер, время отправки).

После отправки значительного количества пакетов каждый процессор синхронизирует свой счетчик со счетчиком шлюза по следующему правилу:

- если значение счетчика шлюза больше значения счетчика процессора, то надо

заменить значение счетчика процессора значением счетчика шлюза;

- если же значение счетчика процессора больше значения счетчика шлюза, то надо заменить значение счетчика шлюза значением счетчика процессора.

Программный модуль приемника принимает пакет, проверяет целостность метаданных с помощь имитовставки. Если значение имитовставки, рассчитанное в компьютере-приемнике, не совпадает с содержащимся в пакете значением, то пакет отклоняется. Если значение имитовставки совпало, то выполняется проверка на повтор.

Проверка на повтор принятого пакета, состоит из следующих действий:

- если время отправки принятого пакета выходит за пределы промежутка времени рассогласования, то отклоняют пакет;
- если время отправки пакета находится в пределах промежутка времени рассогласования, то
  - сравнивают время отправки принятого пакета с временем отправки последнего принятого пакета;
  - если время отправки принятого пакета больше времени отправки последнего принятого пакета, то
    - заменяют время отправки последнего принятого пакета на время отправки принятого пакета;
  - если номер принятого пакета NU меньше (сравнение производится по правилам арифметики, без учета сложной структуры поля) любого из номеров ранее принятых от соответствующего отправителя пакетов, то отклоняют пакет;
  - если пакет с номером, соответствующим номеру принятого пакета, уже принимался от соответствующего отправителя, то отклоняют пакет;
  - если номер принятого пакета больше меньшего из номеров ранее принятых от соответствующего отправителя пакетов и меньше наибольшего, то
    - помечают, что от соответствующего отправителя получен пакет с данным номером;
  - если номер принятого пакета больше любого из номеров ранее принятых от соответствующего отправителя пакетов, то
    - номер принятого пакета запоминается как номер последнего принятого от соответствующего отправителя пакета и заносится в список номеров принятых пакетов, а сам список укорачивается;
  - принимают пакет.

В результате за счет контроля времени отправки и номера пакета устраняется возможность повторного принятия пакета, а также достигается дополнительный технический результат:

- необходимость двухсторонней фиксации факта создания соединения отсутствует, так как синхронизация номеров пакетов осуществляется с использованием времени отправки пакета;
- необходимость передавать получателю пакетов информации о количестве реальных шлюзов безопасности отсутствует, так как в номера пакетов формируются таким образом, чтобы множества номеров, назначаемых этим шлюзам, не пересекались, согласно (1);
- необходимость ведения нескольких счетчиков для одного потока информации на стороне принимающего устройства также отсутствует вследствие разделения множеств формируемых шлюзами безопасности номеров пакетов, согласно (1);
- пакеты на стороне шлюза безопасности формируются на максимальной для оборудования скорости, так как у каждого процессора есть собственный счетчик отправляемых пакетов, и процессорам нет необходимости конкурировать за доступ к

единому счетчику.

Причем технический результат достигаются и при наличии единственного, включающего в себя несколько процессоров шлюза, отправляющего пакеты.

Необходимо отметить, что возможны и другие варианты реализации предложенного способа, отличающиеся от описанного выше и зависящие от личных предпочтений при программировании отдельных действий и функций.

(57) Формула изобретения

Способ предотвращения повторного использования пакетов цифровых данных в сетевой системе передачи данных, причем система содержит

несколько шлюзов безопасности, выполненных с возможностью формировать пакеты и имеющих нескольких процессорных блоков, причем каждый процессорный блок имеет отдельный счетчик отправленных пакетов;

компьютер, принимающий сообщения и связанный с другими компьютерами через сеть передачи данных;

способ, заключающийся в том, что

получают в выбранном шлюзе полезные данные для каждого отправляемого пакета;

формируют метаданные для каждого отправляемого пакета, причем метаданные включают

номер пакета, вычисляемый по формуле

$$NU = CP * 2^{NS + NG} + NT * 2^{NG} + NA,$$

где CP - значение счетчика отправляемых пакетов, независимо используемого данным процессором выбранного шлюза безопасности,

NS - количество двоичных разрядов, требуемое для хранения номера процессора в выбранном шлюзе безопасности,

NG - количество двоичных разрядов, требуемое для хранения номера шлюза безопасности,

NT - номер процессора выбранного шлюза безопасности, формирующего данный пакет,

NA - номер шлюза безопасности, формирующего данный пакет,

причем

$$NS = \lceil \log_2 NV \rceil,$$

где NV - общее количество процессоров в данном шлюзе безопасности, осуществляющем защиту трафика,

$\lceil x \rceil$ - операция, возвращающая наименьшее целое число, большее или равное указанному числу x;

$$NG = \lceil \log_2 NB \rceil,$$

где NB - общее количество шлюзов безопасности, осуществляющих защиту трафика;

время отправки пакета данных;

данные для проверки целостности метаданных;

отправляют пакет из выбранного шлюза через сеть передачи данных;

устанавливают на компьютере, принимающем сообщения, допустимую величину промежутка времени рассогласования;

формируют в памяти компьютера, принимающего сообщения, области для хранения следующих данных от каждого отправителя:

времени отправки последнего принятого пакета,

номера последнего принятого пакета,

списка номеров ранее принятых пакетов;

принимают пакет, включающий полезные данные и метаданные;

проверяют целостность метаданных принятого пакета, используя данные для проверки целостности метаданных;

проводят проверку на повтор принятого пакета, выполняя следующие действия с использованием хранимых в памяти данных для данного отправителя:

если время отправки принятого пакета выходит за пределы промежутка времени рассогласования, то отклоняют пакет;

если время отправки пакета находится в пределах промежутка времени рассогласования, то:

если время отправки принятого пакета больше времени отправки последнего принятого пакета, то

заменяют время отправки последнего принятого пакета на время отправки принятого пакета;

заменяют номер последнего принятого пакета на номер принятого пакета;

заносят номер принятого пакета в список номеров ранее принятых пакетов;

принимают пакет;

если время отправки принятого пакета не превышает времени отправки последнего принятого пакета, то:

если номер последнего принятого пакета меньше номера принятого пакета, то

заменяют номер последнего принятого пакета на номер принятого пакета;

заносят номер принятого пакета в список номеров ранее принятых пакетов;

принимают пакет;

если номер последнего принятого пакета больше или равен номеру принятого пакета, то

проводят поиск номера принятого пакета в списке номеров ранее принятых пакетов;

если номер принятого пакета найден в списке номеров ранее принятых пакетов, то

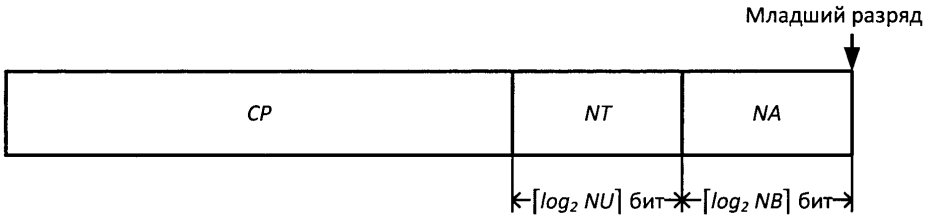
отклоняют принятый пакет;

если номер принятого пакета не найден в списке номеров ранее принятых пакетов, то

заносят номер принятого пакета в список номеров ранее принятых пакетов;

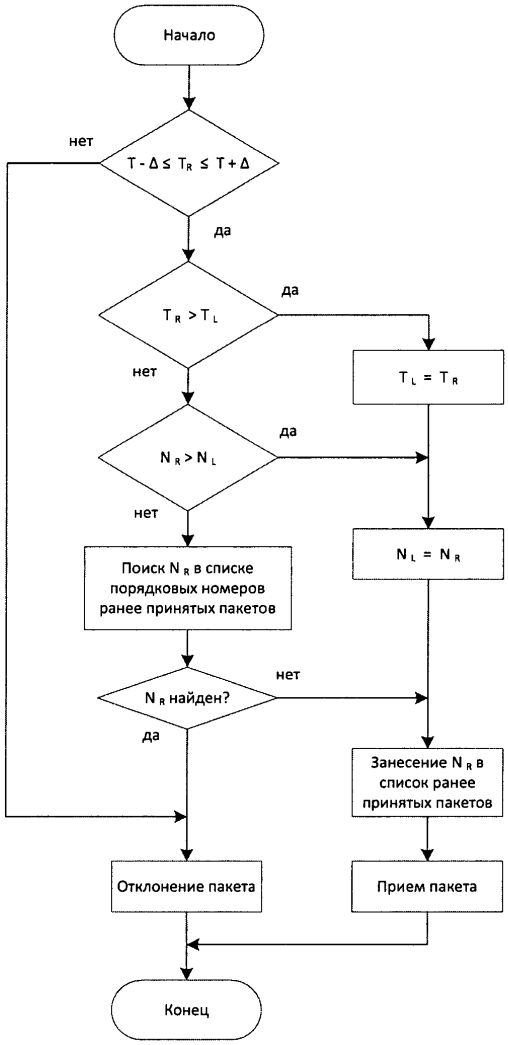
принимают принятый пакет.

1



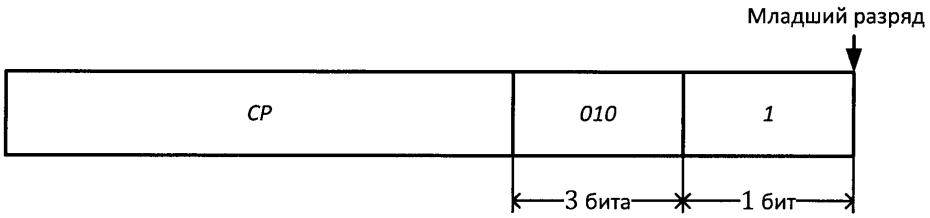
Фиг. 1

2


Начало
нет
$T-\Delta \le T_R \le T+\Delta$
да
$T_R > T_L$
да
$T_L = T_R$
нет
$N_R > N_L$
да
нет
$N_L = N_R$
Поиск N_R в списке порядковых номеров ранее принятых пакетов
N_R найден?
нет
да
Занесение N_R в список ранее принятых пакетов
Отклонение пакета
Прием пакета
Конец

Фиг. 2

3



Фиг. 3