УДК 621.327:629.391

С.В. Дуденко, С.В. Алексеев, А.В. Перепелица

Харьковский университет Воздушных Сил им. И. Кожедуба

СПОСОБ ФОРМИРОВАНИЯ УНИКАЛЬНЫХ ИДЕНТИФИКАТОРОВ ОБЪЕКТОВ ДЛЯ ПОДСИСТЕМ РЕПЛИКАЦИИ ДАННЫХ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ ОБРАБОТКИ ЦИФРОВОЙ ИНФОРМАЦИИ

В статье предложен способ формирования уникальных идентификаторов объектов для подсистем репликации (синхронизации) данных в распределенных системах обработки цифровой информации на основе алгоритма хеширования. Рассмотренные алгоритм хеширования и предложенный способ при верном выборе исходных параметров обеспечат идентификацию объектов и существенно сократят объем служебной информации.

репликация данных, алгоритм хеширования, уникальный идентификатор объекта

Введение

Постановка задачи. Способы обработки информации с использованием вычислительной техники развиваются быстрыми темпами, что приводит к увеличению объемов данных, циркулирующих в компьютерных сетях и системах. При этом остается актуальной задача обеспечения репликации (синхронизации) данных, хранимых на разных вычислительных узлах. Данная задача решается на основе использования различных сценариев репликации с применением глобальных уникальных идентификаторов (GUID).

Стандартный алгоритм построения глобальных уникальных идентификаторов формирует значения GUID размером 36 байт и имеет формат вида: 7E9FS2B4-87B8-448C-AB23-E9A2C8763QA9. Он обеспечивает уникальность в мировом масштабе, но имеет сравнительно большой размер GUID, что приводит к увеличению сетевого трафика.

Предположим, что на узлах распределенной системы размещена информация о 10 000 объектах, имеющих характеристики: название и координаты объекта. При синхронизации информации между двумя узлами и фиксированном объеме характеристик одного объекта в 1 000 байт, полезный объем информации составит 10 000 000 байт.

При использовании для идентификации объектов GUID размером 36 байт (дополнительно 3,6% относительно объема характеристик объекта), служебный объем информации составит 360 000 байт, что при скорости передачи данных по каналу связи в 8 Кбит/с потребует дополнительных 6 минут на передачу, при 80 Кбит/с – 36 секунд.

Для большинства систем передачи данных принято считать, что приемлемый уровень служебной информации не должен превышать 10% пропускной способности канала связи. Тогда, если не учитывать избыточность применяемых протоколов передачи данных, при использовании GUID размером 36 байт пороговым будет объем характеристик одного объекта в 360 байт.

Таким образом, для систем, имеющих необходимость репликации большого количества объектов с характеристиками малого объема, актуальной является задача формирования уникальных идентификаторов объектов меньшего размера с целью уменьшения сетевого трафика.

Анализ литературы. В [1 – 3] рассмотрены функции хеширования ГОСТ Р 34.11-94 и MD5, формирующие хэш-код длиной 128 бит, ГОСТ 34.11-94 и MD4 – 256 бит.

Цель статьи. Введение в рассмотрение и исследование способа формирования уникальных идентификаторов для подсистем репликации (синхронизации) в распределенных системах обработки цифровой информации на основе функции хеширования.

Основной материал

При анализе произвольной распределенной системы обработки информации на первом этапе представляют её в виде связанного графа (рис. 1), в качестве ребер которого используется факт наличия канала связи между узлами системы.

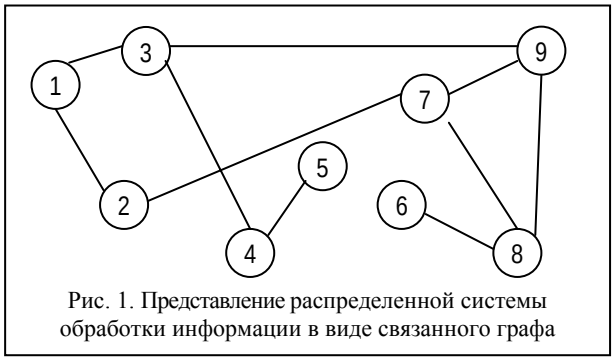

Рис. 1. Представление распределенной системы обработки информации в виде связанного графа

Для идентификации узлов в системе используют различные методы.

Рассмотрим вариант использования для идентификации объектов пары

$$GUID_{узла} + ID_{объекта}, \quad (1)$$

где $GUID_{узла}$ – уникальный идентификатор узла в

системе; $ID_{объекта}$ – уникальный идентификатор объекта в пределах узла.

При таком подходе обеспечивается решение задачи идентификации объектов в пределах системы. Для 10 тысяч объектов на одном узле достаточно $ID_{объекта}$ размером 2 байта.

В реляционных базах данных [4, 5] для автоматического формирования $ID_{объекта}$ предлагается функция инкремента, которая перестает работать при переполнении счетчика, а также неудобна при частых операциях удаления и создания объектов.

Рассмотрим применение для решения нашей задачи алгоритмов хеширования, используемых при криптографической защите информации.

Алгоритм хеширования.

Хэш-функция h(M) – это однонаправленное отображение сообщения M из множества 2^{l_m} в сообщение N из множества 2^{l_h}, где l_m – размер сообщения. Другими словами N - сжатый образ сообщения M. Например в битах: $2^{10^6} \xrightarrow{h} 2^{256}$. В качестве h(M) в криптографии можно использовать MD4, MD5, ГОСТ 34.11-94 и другие. Но данные функции с целью обеспечения криптостойкости имеют увеличенный размер хэш-кода, как правило, 128 бит и более, что не подходит для наших целей.

В [3] приведено два простейших алгоритма формирования хэш-кода, основанные на операциях побитового ИСКЛЮЧАЮЩЕГО ИЛИ и сдвига, которые могут быть использованы, но оставлен открытым вопрос выбора исходных сообщений M, способ выбора которых будет значительным образом влиять на хэш-коды N.

Рассмотрим вариант формирования исходного сообщения M как:

$$\text{Timestamp} \copyright GUID_{узла} \copyright \text{Random}, \quad (2)$$

где Timestamp – временная метка, которая обеспечивает уникальность исходного сообщения в пределах узла; © – функция конкатенации значений; $GUID_{узла}$ – уникальный идентификатор узла в системе, который будет обеспечивать разделение пространства значений хэш-кодов N на несколько диапазонов; Random – случайное число, обеспечивающее рассеивание хэш-кодов N по всему диапазону.

Пример исходного сообщения: 61017,39369.75 © 8228-626029686-74-21840 © 280964101 = 354 бита.

Для проведения статистического исследования используем алгоритм хеширования, представленный на рис. 2, где l – длина хэш-кода.

Эксперимент, результаты которого приведены в табл. 1, проводился в системе CACHE 5.0 [5], при условии генерации 100 000 исходных сообщений по правилу (2) с использованием предложенного алгоритма (рис. 2). В табл. 1 представлены значения: $P_{ож}$ – ожидаемая вероятность события, что произойдет повтор хэш-кодов (выбор из одной корзины), $P_{экспер}$ – вероятность того же события на основе статистики. Как видно из таблицы при $l > 56$ $P_{ож} \to 0$, что подтверждается результатами экспериментов.

Использование предложенного алгоритма хеширования (рис. 2) для уникальной идентификации узлов (рис. 1) не рекомендуется в связи с возможностью повторов хэш-кодов.

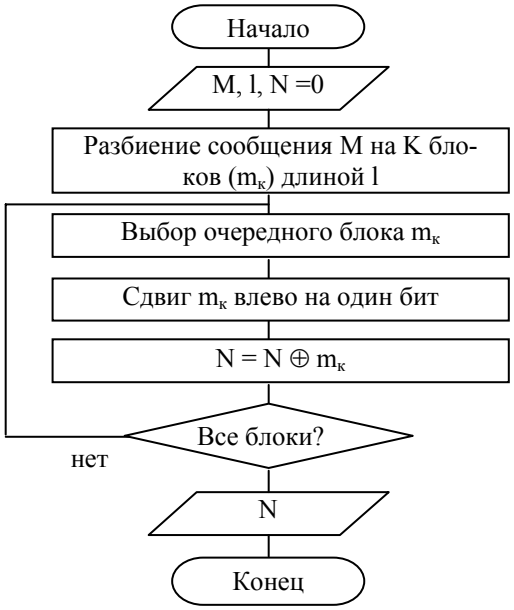

Рис. 2. Предложенный алгоритм хеширования

Таблица 1

Результаты статистического исследования предложенного алгоритма хеширования для формирования идентификаторов

l, бит	Длина идентификатора	Повторов (на 100 000)	$P_{ож}$	$P_{экспер}$
56	~ 127 бит	0	~ 0	~ 0
48	~ 101 бит	4	~ 0	$3 \cdot 10^{-5}$
40	~ 87 бит	30	~ 0	$3 \cdot 10^{-4}$
32	~ 59 бит	4000	~ 0	0,04
24	~ 46 бит	17000	~ 0	0,17
16	~ 31 бит	95000	0,672	0,95
8	~ 8 бит	99980	0,99872	0,999

Следует отметить, что способ формирования случайных чисел оказывает существенное влияние на формируемый хэш-код.

Традиционно при генерировании случайных чисел требуется, чтобы получаемая последовательность была случайной в некотором вполне определенном статистическом смысле [3]. При этом для проверки любой последовательности на случайность обычно применяют два критерия: однородность распределения (частота появления в последовательности конкретного значения должна быть примерно равномерной для всех значений диапазона); независимость (значения последовательности не должны зависеть друг от друга).

Самым популярным алгоритмом генерации псевдослучайных чисел является метод линейного сравнения Лемера [3]. Этот алгоритм имеет четыре параметра: m – модуль сравнения, m>0; a – множитель, $0 \le a < m$; c – приращение, $0 \le c < m$; X_0 – начальное или порождающее число, $0 \le X_0 < m$.

Последовательность случайных чисел $\{X_0\}$ получают с помощью итераций:

$$X_{n+1} = (a \cdot X_n + c) \bmod m. \quad (3)$$

В [6] предложено три критерия для оценки качества любого генератора случайных чисел: генерирующая функция должна быть функцией полного периода; генерируемая последовательность должна вести себя как случайная; генерирующая функция должна эффективно реализовываться в рамках 32-битной арифметики. Все эти критерии могут быть удовлетворены выбором значений a, c и m.

Если m является простым и c=0, для некоторых значений a период генерируемой последовательности оказывается равным m–1 и в этой последовательности отсутствует только значение 0. При $m=2^{31}-1$ генерирующая функция (3) примет вид

$$X_{n+1} = (a \cdot X_n + c) \bmod (2^{31} - 1). \quad (4)$$

Одним из значений параметра a, удовлетворяющим приведенным критериям, является $a = 7^5 = 16807$. Соответствующий генератор находит очень широкое применение и нередко рекомендуется для статистического и имитационного моделирования различных процессов.

Преимуществом алгоритма линейного сравнения является то, что при верном выборе параметров генерируемая последовательность псевдослучайных чисел оказывается статистически неотличимой от последовательности чисел, выбираемых случайно (но безвозвратно) из множества $1, 2, \ldots m-1$.

Рассмотрим способ формирования уникальных идентификаторов по правилу конкатенации хэш-кодов, полученных в соответствии с рассмотренным ранее алгоритмом (рис. 2). Формирование отдельных хэш-кодов происходит независимо:

$$N_1 © N_2 © \ldots © N_j. \quad (5)$$

Эксперимент, проведенный в системе CACHE 5.0, при условии генерации 1 000 000 идентификаторов по правилу (5) показал, что повторений не произошло для различных вариаций (см. табл. 1). Поэтому мы можем прогнозировать вероятность события, что произойдет повтор идентификаторов в части определения вероятности наступления последовательности независимых событий:

$$P_{прогноз} = P_1 \cdot P_2 \cdot \ldots \cdot P_j \quad (6)$$

Условия экспериментов и прогнозируемые вероятности представлены в табл. 2.

Таблица 2

Результаты статистического исследования предложенного способа формирования уникальных идентификаторов

Структура GUID	Длина GUID	$P_{прогноз}$
56_56	~ 271 бит	~ 0
32_56	~ 200 бит	~ 0
32_48	~ 180 бит	10^{-6}
32_40	~ 162 бит	10^{-5}
32_32	~ 150 бит	0,0016
32_24	~ 126 бит	0,0068
16_32_8_16	~ 248 бит	0,0036

Выводы

Рассмотренные алгоритмы хеширования и способ формирования уникальных идентификаторов для подсистем репликации (синхронизации) в распределенных системах обработки цифровой информации при верном выборе исходных параметров обеспечат идентификацию объектов и существенно сократят объем служебной информации.

Список литературы

1. Соколов А.В., Степанюк О.М. Методы информационной защиты объектов и компьютерных сетей. – М.: ООО «Фирма «Издательство АСТ»; С.-Пб ООО «Издательство «Полигон», 2000. – 272 с.

2. Чмора А.Л. Современная прикладная криптография. – М.: Гелиос АРВ, 2001. – 256 с.

3. Столлингс В. Криптография и защита сетей: принципы и практика. – М.: Вильямс, 2001. – 672 с.

4. Постреляционная СУБД CACHE 5. Объектно-ориентированная разработка приложений. 2-е изд., перераб. и дополн. – М.: ООО "Бином-Пресс", 2005. – 416 с.

5. Конноли Томас. Базы данных: проектирование, реализация и сопровождение. Теория и практика, 2-е изд.: Пер. с англ.: Учебн. пос. – М.: Издательский дом "Вильямс", 2000. – 1120 с.

6. Park S., Miller K. Random Number Generators: Good Ones and Hard to Find // Communications of the ACM. – October 1988. – P. 132-138.

Поступила в редколлегию 24.01.2008

Рецензент: д-р физ.-мат. наук, проф. С.В. Смеляков, Харьковский университет Воздушных Сил им. И. Кожедуба, Харьков.

СПОСІБ ФОРМУВАННЯ УНІКАЛЬНИХ ІДЕНТИФІКАТОРІВ ОБ'ЄКТІВ ДЛЯ ПІДСИСТЕМ РЕПЛІКАЦІЇ ДАНИХ В РОЗПОДІЛЕНИХ СИСТЕМАХ ОБРОБКИ ЦИФРОВОЇ ІНФОРМАЦІЇ

Дуденко С.В., Алексєєв С.В., Перепелиця О.В.

Запропоновано спосіб формування унікальних ідентифікаторів об'єктів для підсистем реплікації (синхронізації) даних в розподілених системах обробки цифрової інформації на базі алгоритму хешування. Розглянуті алгоритм хешування і запропонований спосіб при вірному виборі початкових параметрів забезпечать ідентифікацію об'єктів і істотно скоротять об'єм службової інформації.

Ключові слова: *реплікація даних, алгоритм хешування, унікальний ідентифікатор об'єкта.*

FORMING METHOD OF UNIQUE IDENTIFIERS OF OBJECTS FOR REPLICATION SUBSYSTEMS OF INFORMATION IN THE DISTRIBUTED SYSTEMS OF DIGITAL INFORMATION TREATMENT

Dudenko S.V., Alekseev S.V., Perepelitsa A.V.

The forming method of unique identifiers of objects is offered for the subsystems of replication (synchronization) of information in the distributed systems of digital information treatment on the basis of hashing algorithm. Considered a hashing algorithm and offered method at the faithful choice of initial parameters will provide authentication of objects and substantially will shorten the volume of service information.

Keywords: *replication information, hashing algorithm, unique identifiers of objects.*