

ПОИСК ДОМЕННЫХ ИМЕН В КОРПОРАТИВНЫХ СЕТЯХ С ИСПОЛЬЗОВАНИЕМ РЕШЕТЧАТЫХ МОДЕЛЕЙ

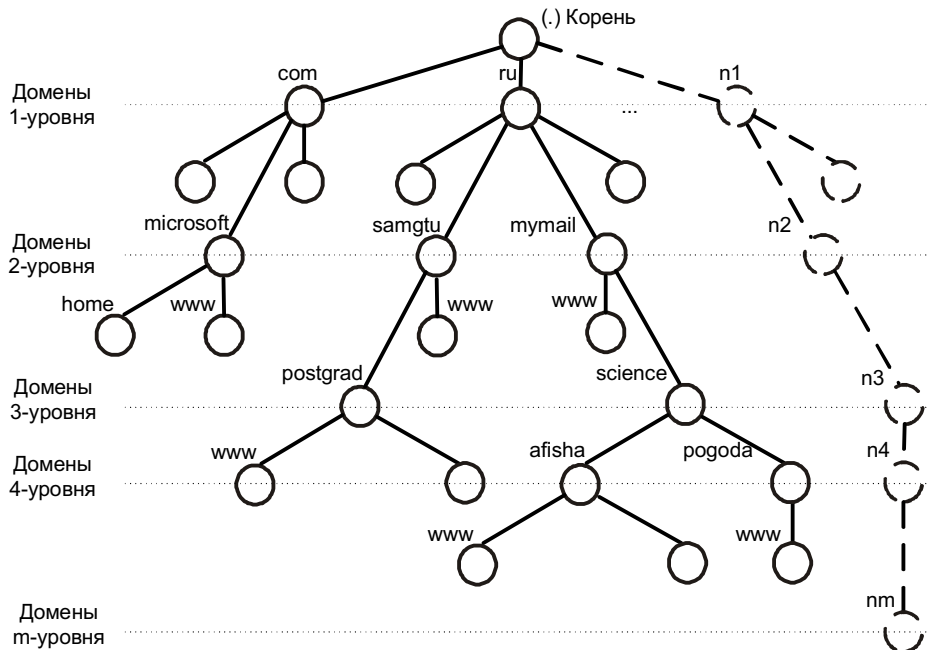
А.С. Храмцев

Самарский государственный технический университет
443100, г. Самара, ул. Молодогвардейская, 244

Статья посвящена применению решетчатых моделей при построении доменных имен. Задача повышения производительности поиска IP-адреса по доменному имени в корпоративных сетях решается за счет использования структурного формата, в котором доменное имя является элементом решетчатой модели.

Ключевые слова: домен, поддомен, DNS-сервер, DNS-клиент, разрешение имени, решетка.

Введение. В любой крупной корпоративной сети используется доменная система имен, состоящая из произвольного количества составных частей. Иерархия доменных имен аналогична иерархии имен файлов. Дерево имен начинается с корня, обозначаемого точкой (.). Запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяются друг от друга точкой. Совокупность имен с одинаковыми старшими составными частями образуют домен имен [1]. Рассмотрим пример доменной системы имен, иерархическая древовидная структура которой представлена на рис. 1.



Р и с. 1. Пространство доменных имен

Различают домены первого, второго, третьего и т. д. уровня. Например, имена `www.samgtu.ru` и `www.mymail.ru` входят в домен первого уровня `ru` (имеют одну общую старшую часть `ru`). Имя `science.mymail.ru` входит в домен второго уровня `mymail.ru` (имеет две старшие части `mymail.ru`). Имена `afisha.science.mymail.ru`, `rogoda.science.mymail.ru` являются поддоменами домена `science.mymail.ru`. При задании в каждом домене и поддомене уникальных имен следующего уровня иерархии создается система из уникальных имен.

Централизованная служба системы доменных имен (DNS), основанная на распределенной базе отображений «доменное имя – IP-адрес», использует в своей работе DNS-серверы и DNS-клиенты. DNS-сервер – специализированное программное обеспечение (ПО) для обслуживания DNS, а также компьютер, на котором это ПО выполняется. DNS-клиент – специализированная библиотека или программа для работы с DNS. DNS-серверы поддерживают распределенную базу отображений, а DNS-клиенты обращаются к серверам с запросами о поиске IP-адреса узла по доменному имени.

При росте количества узлов в сети задача масштабирования решается созданием новых доменов и поддоменов имен и добавлением в службу DNS новых серверов. Существует два распределения имен на серверах. В первом случае DNS-сервер хранит отображения «доменное имя – IP-адрес» для всего домена и всех его поддоменов. Однако такой подход повышает нагрузку на сервер. Другой подход состоит в том, что DNS-сервер хранит только отображения, которые заканчиваются на следующем ниже уровне иерархии по сравнению с именем домена. При таком подходе нагрузка распределена между всеми DNS-серверами сети. Помимо таблицы отображений имен каждый DNS-сервер содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS [1].

Под процедурой разрешения DNS-имени подразумевается процедура поиска IP-адреса узла по доменному имени. Существуют нерекурсивная и рекурсивная процедуры разрешения DNS-имен.

В первом случае (нерекурсивная процедура) поиск IP-адреса координирует DNS-клиент. DNS-клиент выполняет запрос к корневому DNS-серверу с указанием полного доменного имени. Корневой DNS-сервер возвращает DNS-клиенту IP-адрес DNS-сервера, обслуживающего домен на следующем уровне иерархии запрошенного имени. DNS-клиент выполняет запрос к следующему DNS-серверу, который возвращает IP-адрес DNS-сервера следующего поддомена. Процедура продолжается до тех пор, пока не будет найден DNS-сервер, содержащий запрошенное имя в своей таблице отображений. Этот DNS-сервер передает IP-адрес запрошенного имени DNS-клиенту.

Во втором случае (рекурсивная процедура) DNS-клиент передает запрос по поиску IP-адреса локальному DNS-серверу. Если DNS-сервер содержит запрошенное имя в своей таблице отображений или запрошенное имя хранится в его кэше, то он сразу возвращает IP-адрес DNS-клиенту. Если DNS-сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу и т. д., аналогично предыдущему варианту. Получив IP-адрес запрошенного имени, локальный DNS-сервер передает его DNS-клиенту, который ждет от него ответ.

Существующие схемы разрешения DNS-имен сопряжены с итеративным выполнением последовательности запросов к разным серверам имен, что требует столько запросов, сколько существует уровней иерархии.

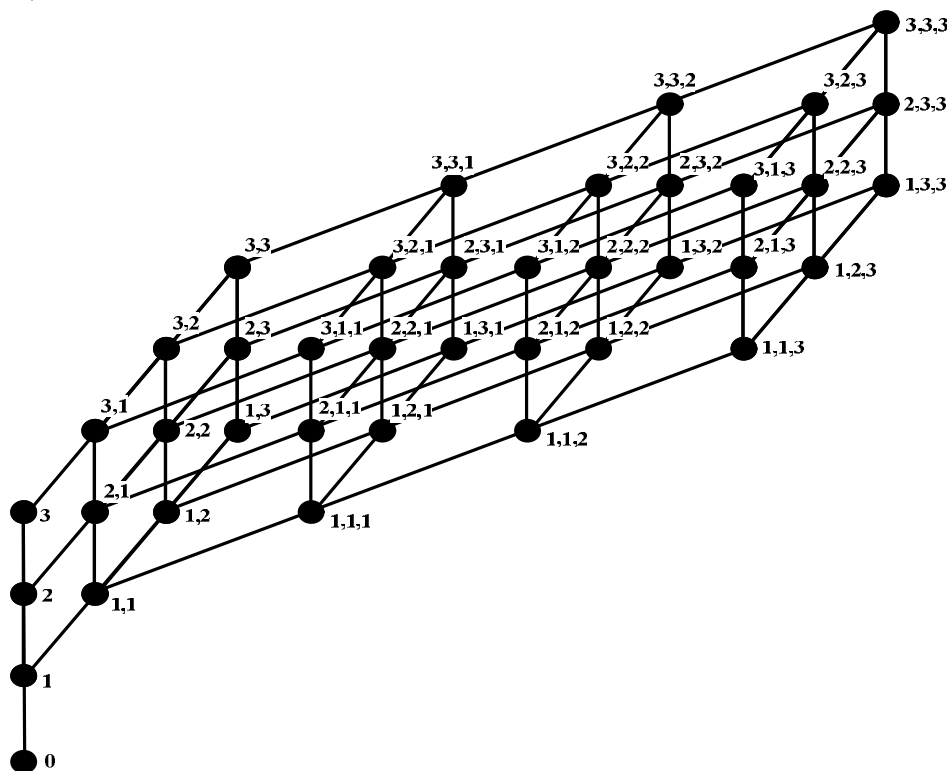
Задача повышения производительности поиска IP-адреса по доменному имени в корпоративных сетях может быть решена за счет использования структурного формата, в котором доменное имя является элементом решетчатой модели.

Структурный метод оптимизации разрешения DNS-имени. Решеткой D называется упорядоченное множество $D = (D, \leq)$, для каждого из подмножеств которого можно определить точные верхнюю и нижнюю грани [2]. Согласно [3] решеткой делителей называется решетка, полученная в результате произведения простых цепей. Простой цепью называется связный граф, в котором все вершины имеют степень 2, за исключением начальной и конечной, которые имеют степень 1. Степень вершины – количество входящих в нее ребер.

Представим доменное имя в виде решетки $D = (n_1, n_2, \dots, n_M)$, где M – количество составных частей доменного имени, n_i – порядковый номер домена i , причем множество всех доменов упорядочено отношением « \leq », означающим, что для любых двух $d_1 \in D$ и $d_2 \in D$, где D – множество всех доменных имен, верно утверждение

$$d_1 \leq d_2 \Leftrightarrow (M_1 \leq M_2) \& (n_1 \leq n_2). \quad (1)$$

Диаграмма подрешетки решетки D изображена на рис. 2. Цифрами в обозначении элемента решетки указаны порядковые номера доменов. Количество составных частей доменного имени соответствует количеству цифр в обозначении элемента решетки.



Р и с. 2. Решетка доменных имен для доменов не более второго уровня с количеством поддоменов на данном уровне не более трех

Задача определения множества путей из 0 в 1 на решетке делителей может возникнуть в связи с определением сложности доменной системы имен. Согласно [2], если D содержит единственный минимальный элемент, то этот элемент называют наименьшим, или 0-элементом; аналогично единственный максимальный элемент называют наибольшим, или 1-элементом. На рис. 2 будем считать 0-элементом вершину, обозначенную как (1), 1-элементом – вершину, обозначенную как (3,3,3). В связи с тем, что множество доменов частично упорядочено и в иерархической структуре нет циклов, введем функцию $f(n_1, n_2, \dots, n_M)$ подсчитывающую множество путей на решетке делителей D , рекурсивное определение которой представлено как

$$f(n_1, n_2, \dots, n_M) := \begin{cases} 1 & \text{if } M=1 \\ (n_1+1) & \text{if } (M=2) \wedge (n_M=1) \\ f(n_1, n_2, \dots, n_{M-1}) & \text{if } n_M=0 \\ \sum_1^b (f(a_1, a_2, \dots, a_{M-1}) \cdot f(\bar{a}_1, \bar{a}_2, \dots, \bar{a}_{M-1})) & \end{cases}, \quad (2)$$

где $b = \prod_{i=1}^{M-1} (n_i + 1)$; a_i – принимает все значения от 0 до n_M ; $\bar{a}_i = (n_M - a_i)$ – дополнение элемента a_i .

Согласно [4] формулу, которая задает произвольную решетку D делителей, называемую также решеткой Юнга, можно представить как

$$f(n_1, n_2, \dots, n_M) = \frac{h(D_j) + 1}{n_M + 1} \cdot f(n_1, n_2, \dots, n_M), \quad (3)$$

где $h(D_j) = \sum_{i=1}^M n_i$ – высота элемента D_j решетки D .

Подрешетка, для которой заданы параметры $M \leq 3$ и $n_M \leq 3$, изображена на рис. 2. Используя формулу (3), получим значение функции f (формула (2)), определяющей множество путей для достижения требуемой вершины из заданной:

$$f(n_1, n_2, \dots, n_M) = \frac{\left(\sum_{i=1}^M n_i \right)!}{\prod_{i=1}^M (n_i)!}. \quad (4)$$

Введем ограничения: DNS-сервер хранит таблицу отображений имен, ссылки на DNS-серверы своих поддоменов, ссылки на DNS-серверы доменов верхнего уровня, ссылки на DNS-серверы своего подмножества. Например, элемент (1,1) на рис. 2 в таблице отображений хранит имя (1,1), содержит ссылки на DNS-серверы своих поддоменов (1,1,1), (1,1,2), (1,1,3), ссылки на DNS-серверы верхнего уровня (1), (0), ссылки на DNS-серверы своего подмножества (2,1), (3,1), (1,2), (1,3).

Рассмотрим задачу разрешения доменных имен по существующей схеме разрешения DNS-имен и с использованием структурного формата для имен (1,1,1); (2,1); (2,2,3), если DNS-клиент обращается к DNS-серверу (1,1,3) (см. рис. 2).

Последовательность запросов при разрешении указанных доменных имен по

существующей схеме разрешения DNS-имен приведена в табл. 1, при использовании структурного формата – в табл. 2.

Из анализа последовательностей запросов, приведенных в табл. 1 и табл. 2, следует, что при использовании решетчатых моделей количество запросов по сравнению с количеством запросов существующей схемы разрешения DNS-имен уменьшается, что приводит к сокращению времени на поиск IP-адреса доменного имени при использовании структурного подхода.

Таблица 1

Существующая схема разрешения DNS-имени

Начальная вершина	Последовательность запросов	Конечная вершина
(1,1,3)	→(0)→(1)→(1,1)→	(1,1,1)
(1,1,3)	→(0)→(2)→	(2,1)
(1,1,3)	→(0)→(2)→(2,2)→	(2,2,3)

Таблица 2

Схема разрешения DNS-имен при использовании структурного формата

Начальная вершина	Последовательность запросов	Конечная вершина
(1,1,3)	→(1,1)→	(1,1,1)
(1,1,3)	→(1,1)→	(2,1)
(1,1,3)	→(2,1,3)→	(2,2,3)

Необходимо отметить, что рассматриваемый метод позволяет добавлять несколько доменов одновременно. Для этого DNS-сервер, на котором происходит добавление поддомена, отслеживает, к какому элементу в его подмножестве имен будет происходить добавление. Если добавляемое имя образует новый уровень доменов или исчерпан диапазон номеров поддоменов, то DNS-сервер передает сообщение о переходе от одной структуры к другой и о необходимости перенумерации элементов корневого серверу (элемент (0) на рис. 2). После этого DNS-сервер блокирует элемент решетки, к которому будет добавляться поддомен, на время добавления к нему нового поддомена. Во втором случае DNS-сервер добавляет имя в рамках существующей структуры и блокирует элемент решетки на время добавления к нему нового поддомена. Поскольку переход от одной структуры к другой занимает определенное время, то перенумерацию элементов новой структуры можно осуществлять во время снижения активности пользователей корпоративной сети, например, в обед или ночью.

Заключение. Таким образом, по сравнению с существующими схемами разрешения DNS-имен, сопряженных с итеративным выполнением последовательности запросов к разным серверам имен, что требует столько запросов, сколько существует уровней иерархии, использование решетчатых структур DNS-имен позволяет сократить количество запросов на основе соответствия DNS-имени с решеткой доменных имен. Обновление решетки DNS-имен происходит по мере изменения структуры сети, причем нагрузка на DNS-серверы распределяется между всеми DNS-серверами сети.

Применение решетчатых моделей для построения распределенной базы отображений «доменное имя – IP-адрес» в корпоративных сетях позволяет сократить время на поиск всех старших и младших составляющих доменного имени, а также на поиск доменных имен.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд. – СПб.: Питер, 2006. – 958 с.
2. Биркгоф Г. Теория решеток. – М.: Наука, 1984. – 568 с.
3. Айгнер М. Комбинаторная теория. – М.: Мир, 1982. – 558 с.

4. *Кистанов А.М., Орлов С.П.* Наглядный комбинаторный анализ информационных транзакционных систем. – Самара: СНЦ РАН, 2008. – 206 с.

Статья поступила в редакцию 24 октября 2010 г.

UDC 519.1

**SEARCH DOMAIN NAMES IN CORPORATE NETWORKS
USING LATTICE MODELS**

A.S. Khramtcev

Samara State Technical University
244, Molodogvardeyskaya st., Samara, 443100

The article deals with the usage of lattice models in the construction of domain names. The task of improving IP-addresses search performance for the domain name in corporate networks is solved by using a structural format in which the domain name is an element of a lattice model.

Keywords: domain, subdomain, DNS-server, DNS-client, name resolution, lattice.