

Статическая балансировка серверов в сетях центров обработки данных

Э.А. Гайфутдинов, Московский технический университет связи и информатики (МТУСИ), аспирант; Estu.d.379@mail.ru
 Е.А. Саксонов, МТУСИ, профессор, д.т.н.; e.a.saksonov@mtuci.ru

УДК 004.75

DOI: 10.34832/ELSV.2026.78.4.003

Аннотация. Представлены результаты анализа метода статической взвешенной циклической балансировки (WRR), которые могут быть использованы для балансировки нагрузки серверов в центрах обработки данных. Для реализации алгоритма балансировки используется поллинг-цикл опроса, который позволяет обеспечить требуемые частоты подключения к серверам без конфликтов. Показано, что при использовании поллинг-цикла необходимо учитывать ошибки, связанные с целыми значениями параметров цикла. Полученные результаты могут быть полезны сетевым администраторам центров обработки данных при выборе серверов и распределении запросов между ними.

Ключевые слова: центр обработки данных, статическая балансировка, поллинг-цикл, балансировка серверов, поток запросов.

Для цитирования: Гайфутдинов, Э.А. Статическая балансировка серверов в сетях центров обработки данных / Э.А. Гайфутдинов, Е.А. Саксонов // Электросвязь. – 2026. – № 4. – С. 25-35.

ВВЕДЕНИЕ

Развитие центров обработки данных (ЦОД) обусловило важность повышения эффективности их работы. Одним из направлений повышения эффективности является управление загрузкой серверов. Это связано с высокими затратами при простоях и перегрузке серверов [1–4]. Среди возможных методов управления нагрузкой серверов в распределенных вычислительных средах широко применяется балансировка по заданным параметрам [5–12]. Такой метод не всегда приводит к оптимизации показателей эффективности работы группы серверов, но позволяет гарантировать наиболее значимые показатели их работы.

Из известного многообразия методов балансировки для исследования выбран статический метод — взвешенный циклический алгоритм (Weighted Round Robin, WRR), являющийся развитием простого циклического алгоритма (Round Robin, RR) [13–15]. Метод основан на предварительном определении параметров балансировки и независимости от изменений состояния серверов и потока запросов. Недостатки метода связаны со сложностью оперативной адаптации параметров к указанным изменениям [14, 15].

Однако при большой распространенности указанного метода получено мало результатов, позволяющих количественно определять параметры и оценивать результаты балансировки при различных вариантах формирования цикла балансировки.

Как объект исследования выбрана сеть серверов ЦОД, поскольку она отличается незначительной протяженностью каналов связи, достаточно высокой надежностью и потребностью в настройке при

решении задач обработки запросов от различных источников, а также использует технологию программно-конфигурируемых сетей, позволяющую сегментировать сеть и на программном уровне реализовать предлагаемые методы балансировки [7, 8, 16].

В статье приводятся результаты анализа применения поллинг-цикла для реализации балансировки загрузки серверов. Применение поллинг-цикла обеспечивает отсутствие конфликтов при выборе серверов, обеспечивает однозначные решения при назначении весов для распределения потока запросов между серверами [17–19].

Результаты обобщают и развивают метод статической балансировки.

ОПИСАНИЕ ИССЛЕДУЕМОЙ СИСТЕМЫ

Задача балансировки загрузки серверов (реальных или виртуальных) в ЦОД заключается в распределении поступающих на обслуживание запросов между серверами. Для ее решения требуется провести анализ потоков и процессов обработки запросов.

Исследуется система, состоящая из N серверов, на вход которой поступает поток запросов от R источников. Число приложений, исполняемых по запросам — K . Выполняются условия: $0 < K < \infty$, $0 < N < \infty$, $0 < R < \infty$. Обслуживание запроса требует исполнения одного определенного приложения.

Все серверы, источники и приложения имеют уникальные номера.

Известна матрица исполнения приложений на серверах $Z = \|z_{nk}\|$ ($n=1, 2, \dots, N$; $k=1, 2, \dots, K$), где $z_{nk}=1$, если сервер n может исполнить приложение k , и $z_{nk}=0$, если приложение k не может быть исполнено на сервере n . Элементы Z связаны условиями: