В. М. Ничипорчук

ПОСТРОЕНИЕ ВЕБ-СЛУЖБ В ЗАЩИЩЕННОМ ИСПОЛНЕНИИ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ СИСТЕМ

В современном мире для взаимодействия клиентской и серверной частей прикладных систем посредством открытых каналов практически повсеместно используются веб-службы. Веб-службы обеспечивают платформенную независимость и простоту посредством применения открытых стандартов и протоколов. Передача данных через интернет-протокол HTTP обеспечивает прозрачное взаимодействие клиента и сервера через межсетевые экраны.

Для построения серверной части веб-службы существует два классических архитектурных подхода [1]:

- взаимодействие сервера приложений напрямую с хранилищем данных или СУБД (Рис. 1);
- взаимодействие сервера приложений с Back-end сервером (Рис. 2).

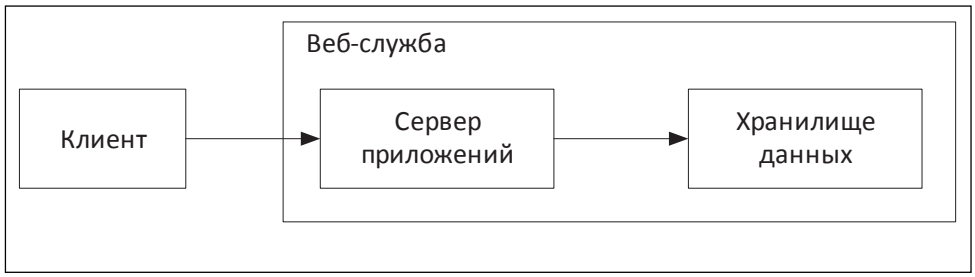

Рис. 1. Веб-служба, сервер приложений напрямую взаимодействует с хранилищем данных

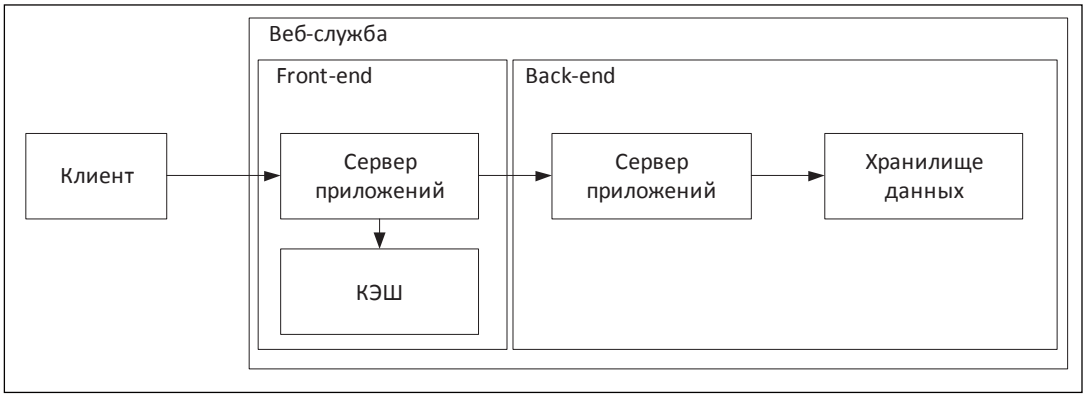

Рис. 2. Веб-служба, взаимодействие Front-end сервера приложений с Back-end сервером

Названные архитектурные подходы к построению веб-служб просты в разработке, для каждого элемента существует множество описанных методик и средств по вертикальному и горизонтальному масштабированию, система легко описывается математической моделью, так как при обычных режимах работы имеет постоянную пропускную способность при обработке сообщений, равную пропускной способности самого медленного компонента.

В то же время при высоких нагрузках увеличивается продолжительность отклика всех компонентов системы и, как следствие, число активных соединений. В результате этого на стороне клиента установленное соединение находится в режиме ожидания, что в ряде стандартных подходов построения клиентских приложений при синхронном взаимодействии приводит к задержке клиента, доставляя эргономический дискомфорт, а также в случае медленных и нестабильных соединений увеличивает вероятность ошибки — обрыв соединения во время ожидания клиента.

На стороне сервера большое количество активных соединений приводит к резкому снижению производительности, в результате чего происходит отказ в обслуживании клиентов. Одной из задач информационной безопасности является обеспечение доступности информации, в таком случае требование доступности может быть нарушено.

Для задач, к которым предъявляются требования обработки интенсивно поступающих сообщений, своевременности обслуживания клиентов (например, прием платежей) и доступности, особенно при наличии нестабильных каналов связи с клиентами, неопределенного времени обработки сообщений (например, взаимодействие со сторонними системами с неизвестным и нестабильным временем отклика), предлагается использование асинхронного подхода [2] к обработке сообщений.

Серверная часть веб-службы включает в себя элементы:

– сервер приложений;

– очередь;

– скоростной кэш;

– сервер обработки сообщений.

Предлагаемая архитектура веб-службы для этого случая в обобщенном виде показана на Рис. 3.

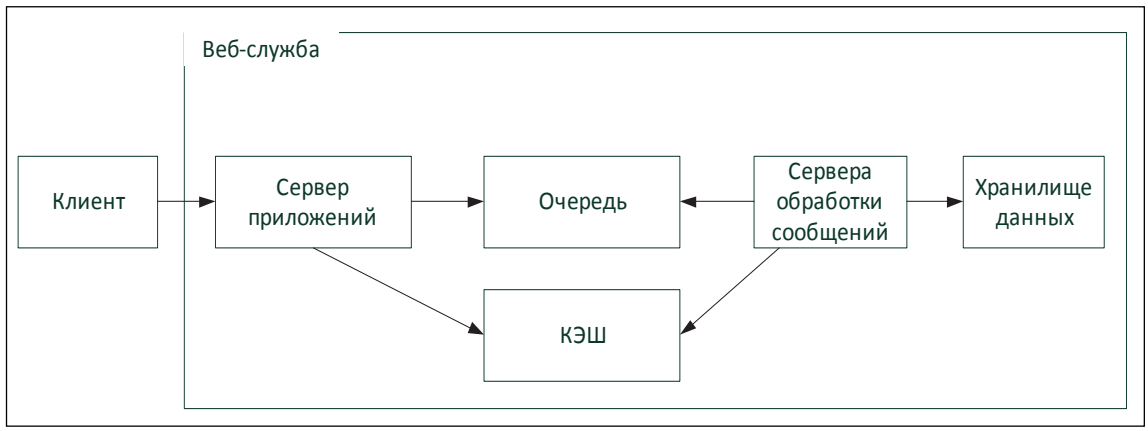

Рис. 3. Предлагаемая архитектура построения веб-службы

При данном подходе алгоритм работы сервера заключается в следующем.

1. Сервер приложений принимает сообщение от клиента, генерирует номер транзакции и производит постановку сообщения и номера транзакции в очередь, далее сервер в ответе клиенту сообщает номер транзакции. Данная операция имеет небольшой временной вес, так как современные реализации серверов, специализированные под формирование и поддержку очереди, функционируют быстрее реляционных СУБД, обремененных дополнительным функционалом и сложностью конфигурации, а также имеют практически линейную возможность горизонтального масштабирования, что позволяет гарантировать стабильно небольшое время приема сообщения.

2. Сообщение забирается сервером обработки сообщений и обрабатывается необходимое время. В современных реализациях серверов и клиентов очереди операция передачи сообщения из очереди к клиенту происходит с минимальными задержками, так как клиент очереди (в нашем случае это сервер обработки сообщений) поддерживает постоянное соединение с сервером очередей, в рамках которого происходит уведомление клиента о поступившем сообщении и возможности его зарезервировать.

3. После обработки сообщения сервер обработки сообщений размещает в скоростном кэше результат обработки.

4. Сервер приложений принимает запрос клиента с информацией о номере транзакции, проверяет наличие результата в кэше и возвращает статус транзакции или результат обработки.

Шаги 1 и 4 алгоритма выполняются для разных клиентов и сообщений независимо от шагов 2 и 3. Этого удалось достичь из-за наличия асинхронной развязки в виде очереди между серверными компонентами системы.

Для взаимодействия с веб-службой алгоритм работы клиента следующий:

1) клиент посылает сообщение серверу, получает номер транзакции;

2) клиент опрашивает сервер через заданный интервал времени, передавая номер интересующей его транзакции. Получает ответ: статус транзакции или результат обработки. При получении результата обработки опрос сервера прекращается, в зависимости от статуса транзакции опрос продолжается или прекращается (статус об ошибке транзакции).

Описанная схема может быть оптимизирована путем оценки среднего времени обработки сообщений и передачи клиенту рекомендуемого интервала опроса сервера.

Полученная схема построения серверной и клиентской частей веб-службы устойчива к высоким нагрузкам, так как время приема сообщений и ответа минимально и в общем случае не зависит от скорости обработки сообщений. Такой подход позволяет использовать решение на медленных и нестабильных каналах соединения, поскольку время активного соединения клиента с сервером минимально и обрыв связи между запросами клиента не является критическим для выполнения транзакции.

С точки зрения масштабирования при предложенной архитектуре построения веб-службы возможно гибкое и контролируемое наращивание производительности системы как при приеме сообщений и предоставлении результатов обработки, так и при обработке сообщений, позволяя регулировать максимально допустимые задержки и время ожидания при обслуживании клиентов. Например, масштабируя серверы приложений, очередь и скоростной кэш, можно увеличить скорость приема сообщений, скорость предоставления результатов обработки и максимальную длину очереди транзакций, но фактическая скорость обработки сообщений останется прежней и при увеличении нагрузки увеличится средняя длина очереди и время ожидания клиента от посылки сообщения до получения результата. И наоборот, масштабируя серверы обработки сообщений, можно увеличить фактическую скорость обработки сообщений, тем самым сокращая длину очереди и, как следствие, время ожидания клиента.

Одним из свойств архитектуры является возможность организации скрытой и защищенной зоны. Поскольку сервер обработки сообщений является инициатором соединений к внешним по отношению к нему сервисам очереди и кэша, возможна установка и настройка межсетевого экрана, осуществляющего только установку исходящих из защищаемой зоны соединений. Со стороны зоны приема сообщений защищаемая зона будет невидимой. Такой подход к организации защищенных зон может позволить усилить безопасность хранилища данных по сравнению со стандартными подходами [3] к построению и защите веб-служб (Рис. 4), когда межсетевой экран ограничивает доступ к серверу приложений, так как скрытая зона фактически накладывается на зону безопасности, организуемую средствами защиты зоны приема сообщений (Рис. 5).

Из недостатков предложенной архитектуры веб-служб можно выделить следующие:

– более сложная реализация по сравнению с классической моделью;

– более сложная математическая модель;

– большие накладные расходы на обработку сообщений.

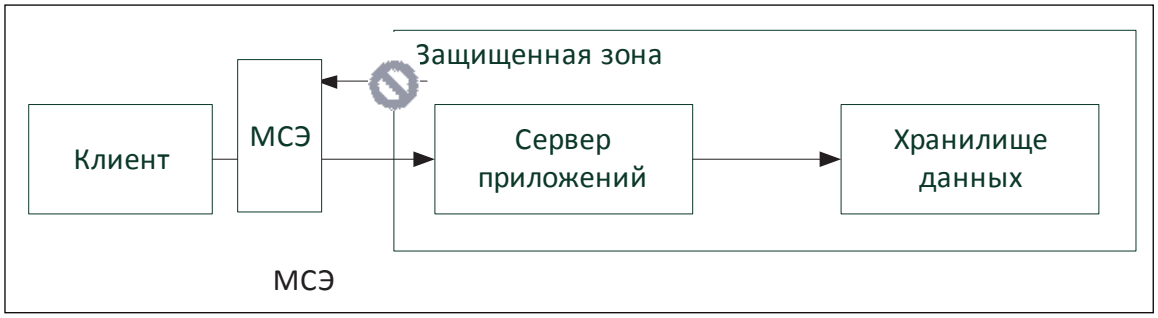

Рис. 4. Стандартный подход защиты и построения веб-службы

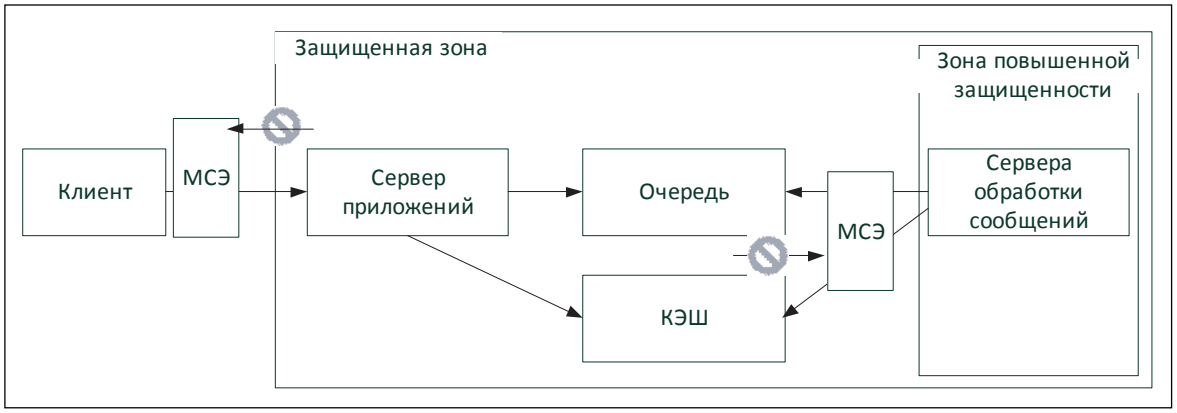

Рис. 5. Предлагаемый подход защиты веб-службы и организации зоны повышенной защищенности

СПИСОК ЛИТЕРАТУРЫ:

1. *Henderson C.* Building Scalable Web Sites. Sebastopol: O'Reilly Media, 2006.
2. *Tanenbaum A. S.* Distributed Systems: Principles and Paradigms, 1st ed. New Jersey: Prentice Hall, 2002. — 840 p.
3. *Запечников С. В., Милославская Н. Г., Толстой А. И., Ушаков Д. В.* Информационная безопасность открытых систем: Учебник для вузов. В 2-х томах. Т. 2. Средства защиты в сетях. М.: Горячая линия—Телеком, 2008. — 558 с.