

Сравнение производительности web-серверов, написанных на платформах Node.js и .NET

Направление подготовки: «090301.65 Компьютерная безопасность»

Выполнил: студент группы КБ-41, Попов Сергей Андреевич

Научный руководитель: д.ф.-м.н., старший преподаватель Власова Ольга Владимировна

В последнее время всё больше и больше разработчиков смотрят в сторону недавно появившейся программной платформы Node.js, главной особенностью которой является событийно-асинхронная архитектура и возможность писать на javascript. Она вызвала небывалый ажиотаж в среде разработчиков, что даже многие стали отказываться от разработки сервисных приложений на тяжеловесной платформе .NET, в пользу Node.js.

Node.js часто хвалят за то, что она отвечает запросам корпораций и позволяет собирать приложения с API, которое может обращаться к серверной части и большим объемам данных в лёгкой и эффективной манере. Действительно, концентрация на переиспользуемом RESTful API как более гибком способе построения архитектуры масштабных программных систем позволила Node.js найти своё место. Некоторые крупные компании вроде PayPal, Yahoo, IBM уже начали использовать возможности Node.js в своих проектах. Даже компания Samsung не так давно опубликовала информацию относительно того, что Node.js и JavaScript работают на низкопроизводительных системах лучше, чем любые другие платформы.

Node.js способен значительно сократить время разработки приложения, сохраняя при этом тот же функционал. Джон Оустерхаут, помогавший в разработке значимого скриптового языка и набора инструментов Tcl/Tk, ещё в 1990-х привёл аргумент о том, что скриптовые языки программирования по своей сути более продуктивны, нежели более тяжеловесные, такие, как C или C++.

По сравнению с более тяжёлыми стеками, такими как .NET, разработка приложений с помощью Node.js происходит быстрее, и с развитием экосистемы Node только ускоряется.

Цели данной работы:

- 1) Изучить принцип работы протоколов транспортного и прикладного уровней и особенности их реализации на платформах .NET и Node.js
- 2) Ознакомиться с асинхронными моделями платформ .NET и Node.js
- 3) Реализовать серверную часть программы для протоколов TCP, UDP, HTTP как на платформе .NET(на языке C#), так и на платформе Node.js(на языке javascript)

- 4) Написать клиентскую часть программы для отправки запросов на сервера на платформах .NET и Node.js
- 5) Выполнить сравнительный анализ производительности с разными настройками серверов на платформах .NET и на Node.js

В данной работе были проведены исследования в области реализации веб-серверов, работающих по протоколам транспортного и прикладного уровней на платформах Node.js и .NET. Были также изучены и различия в асинхронных моделях платформ Node.js и .NET.

Нами исследовалось поведение серверов на платформах Node.js и .NET при трех различных сценариях:

Работа веб-серверов по протоколам транспортного уровня TCP и UDP. Результаты оказались практически идентичными, что свидетельствует о независимости работы реализации протокола от платформы, на которой она написана.

Работа веб-серверов по протоколу прикладного уровня HTTP в многопоточном и однопоточном режимах. В этом эксперименте Node.js значительно опередил .NET, что говорит о том, что его асинхронная модель гораздо лучше спроектирована для работы сервисных приложений.