

Конфігурація веб-серверів в умовах високого навантаження

Волошин Владислав Сергійович

ДА-31

Постановка задачі

- ▶ Об'єкт дослідження:
 - веб-сервер
 - система з високим навантаженням
- ▶ Предмет дослідження:
 - Налаштування веб-сервера для великої кількості запитів та проведення відповідного стрес-тестування

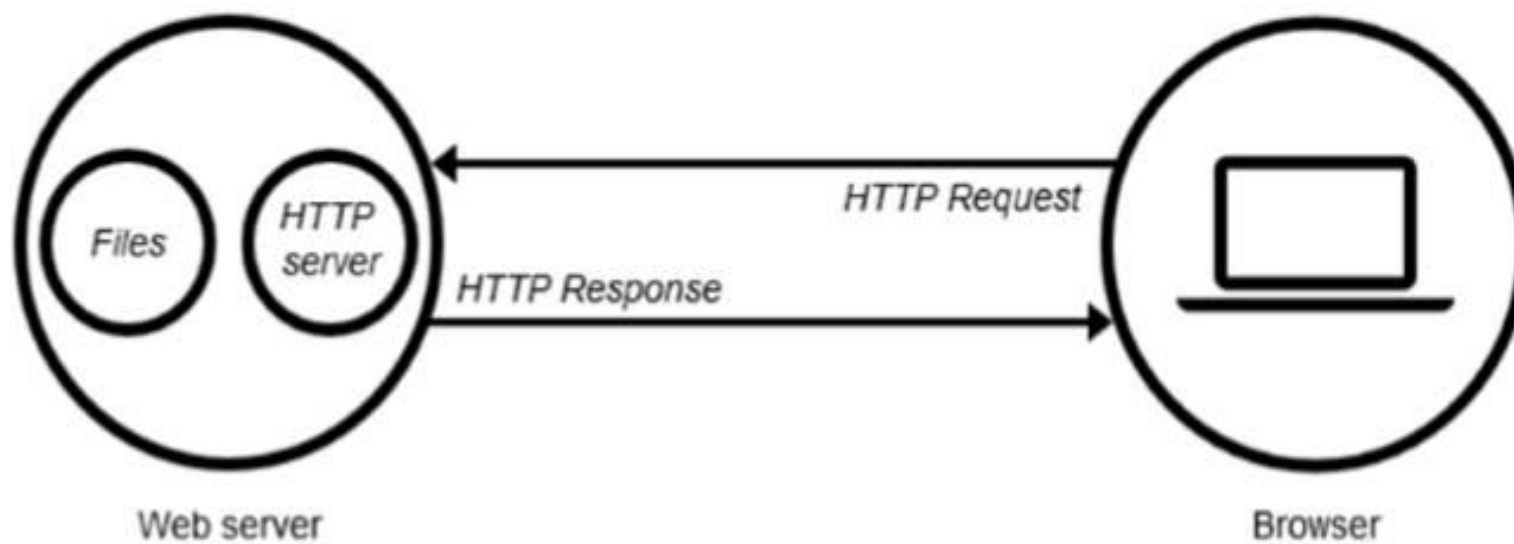
Проблематика

- ▶ Сучасні веб-системи вимагають надійності та достатньої продуктивності для того, аби задовольнити великий попит на них.
- ▶ Більшість сучасних веб-систем користуються в своїй роботі готовими рішеннями в обслуговуванні http-запитів, а саме веб-серверів, їхня конфігурація потребує ретельного аналізу.
- ▶ Конфігурація веб-сервера має бути максимально простою та легкою у підтримці і модернізації. Налаштований веб-сервер має справлятися з навантаженням не менш як 50000 одночасних запитів, не втрачаючи при цьому швидкість обробки запитів.

Основні вимоги до систем з високим навантаженням

- ▶ *Висока доступність та відмовостійкість*
- ▶ *Гнучкість*
- ▶ *Використання простих рішень*
- ▶ *Моніторинг*

Поняття веб-сервера



Способи підвищення продуктивності веб-сервера

- ▶ Конфігурація внутрішніх параметрів веб-сервера
- ▶ Зміни в архітектурі веб-системи

Конфігурація внутрішніх параметрів веб-сервера

- ▶ мінімізація розміру даних передачі
- ▶ мінімізація кількості запитів до серверу
- ▶ оптимізація відображення змісту сайту

Приклади реалізації

- ▶ Gzip
- ▶ HTTP/2
- ▶ Google PageSpeed
- ▶ Мініфікація файлів

Google PageSpeed

- ▶ Додатковий модуль для веб-серверів Apache та Nginx
- ▶ Розроблений компанією Google
- ▶ Модуль стискає статичний контент сайту, запити та оптимізує картинки

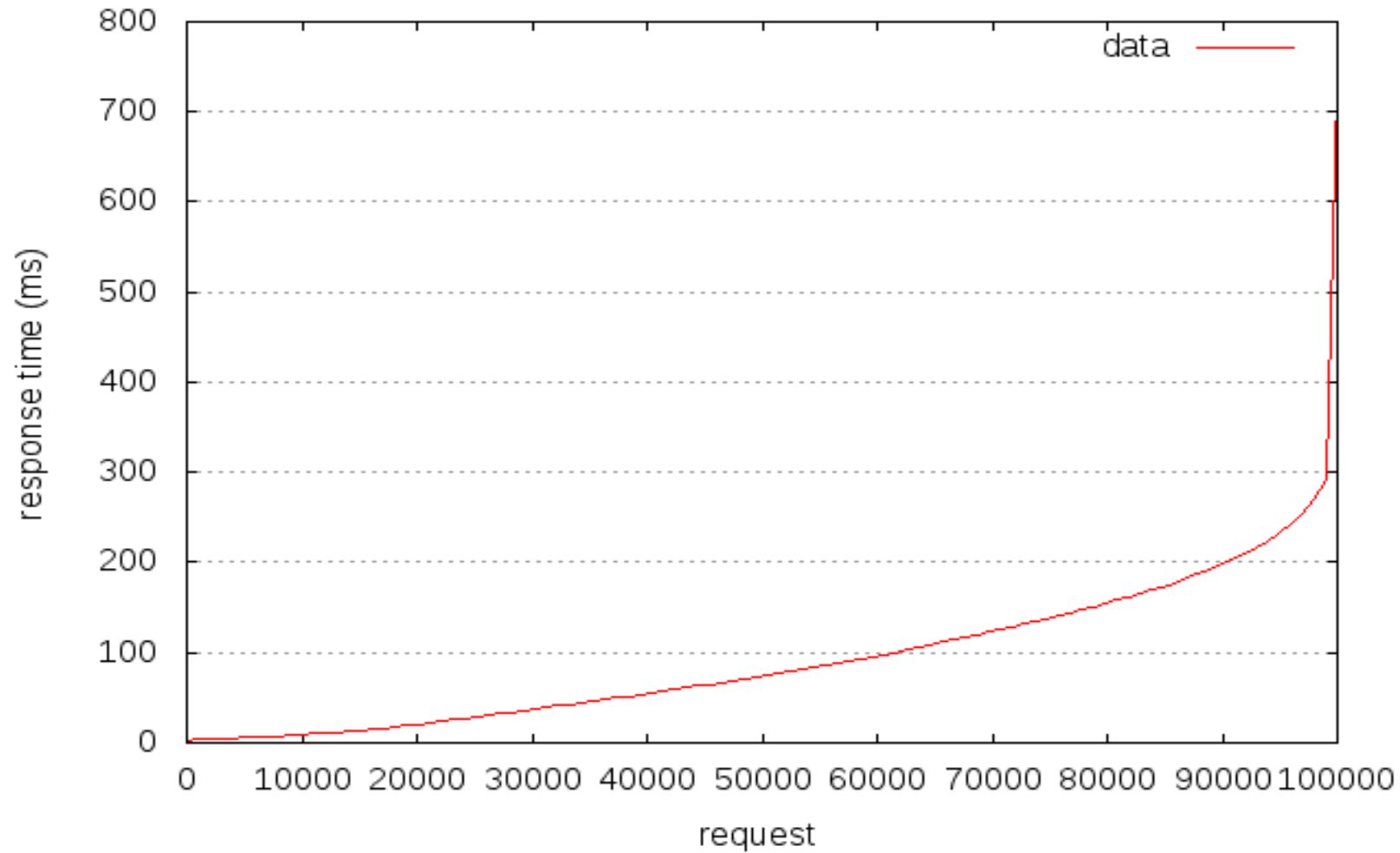
Архітектурні способи підвищення продуктивності веб-серверів

- ▶ Кешування
- ▶ Черги задач
- ▶ Балансування навантаження
- ▶ CDN

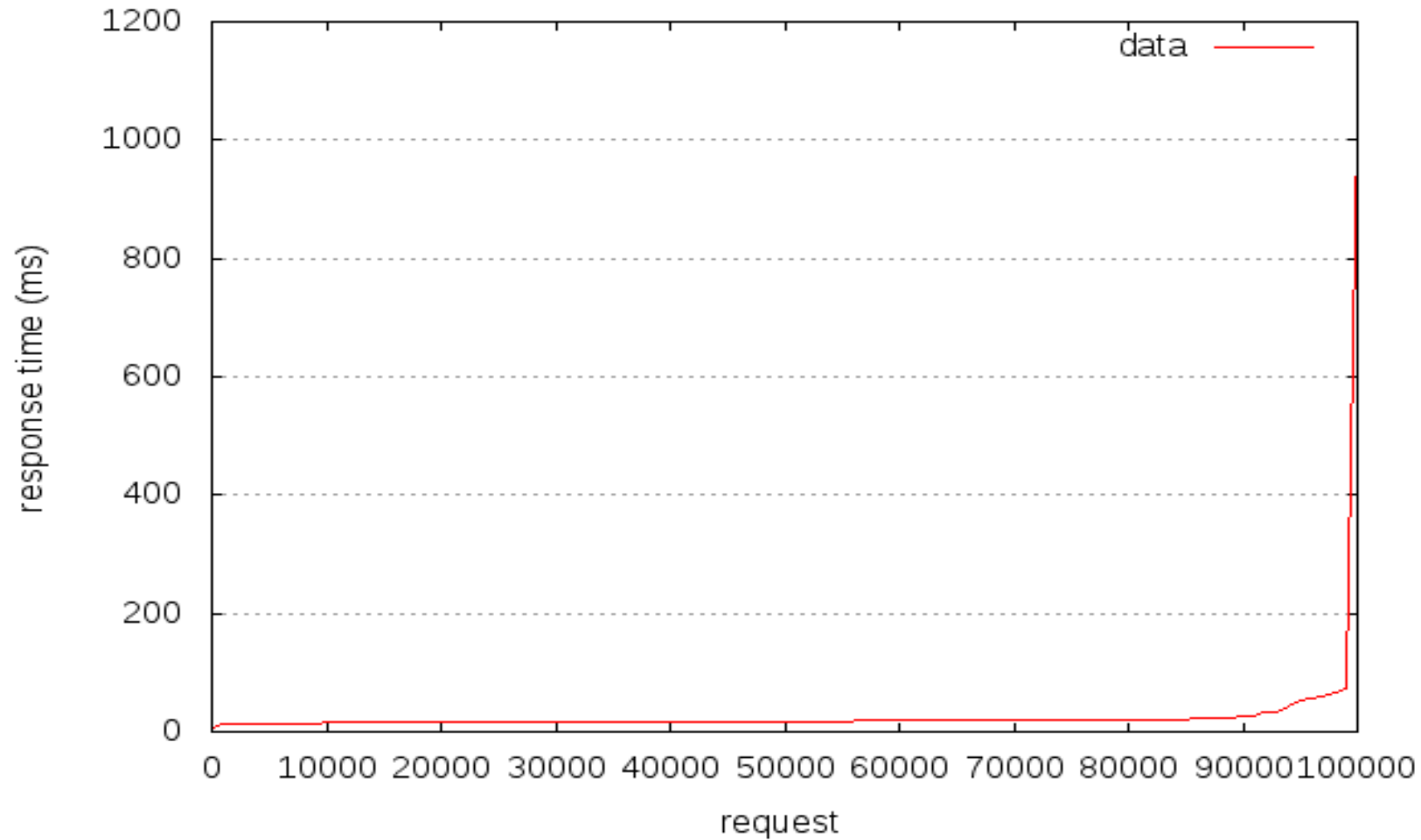
Тестування навантаження на веб-сервер

- ▶ Синтетичні тести Apache Benchmark
- ▶ 2 сервера Ubuntu 14.04 LTS (4ГБ RAM + 4 CPU Cores)
- ▶ Docker-контейнери
- ▶ Веб-сервер Nginx зі статичним контентом
- ▶ 100 000 запитів, які запускаються у 100 потоків

Веб-сервер зі стандартною конфігурацією



Робота веб-сервера з модулем pagespeed

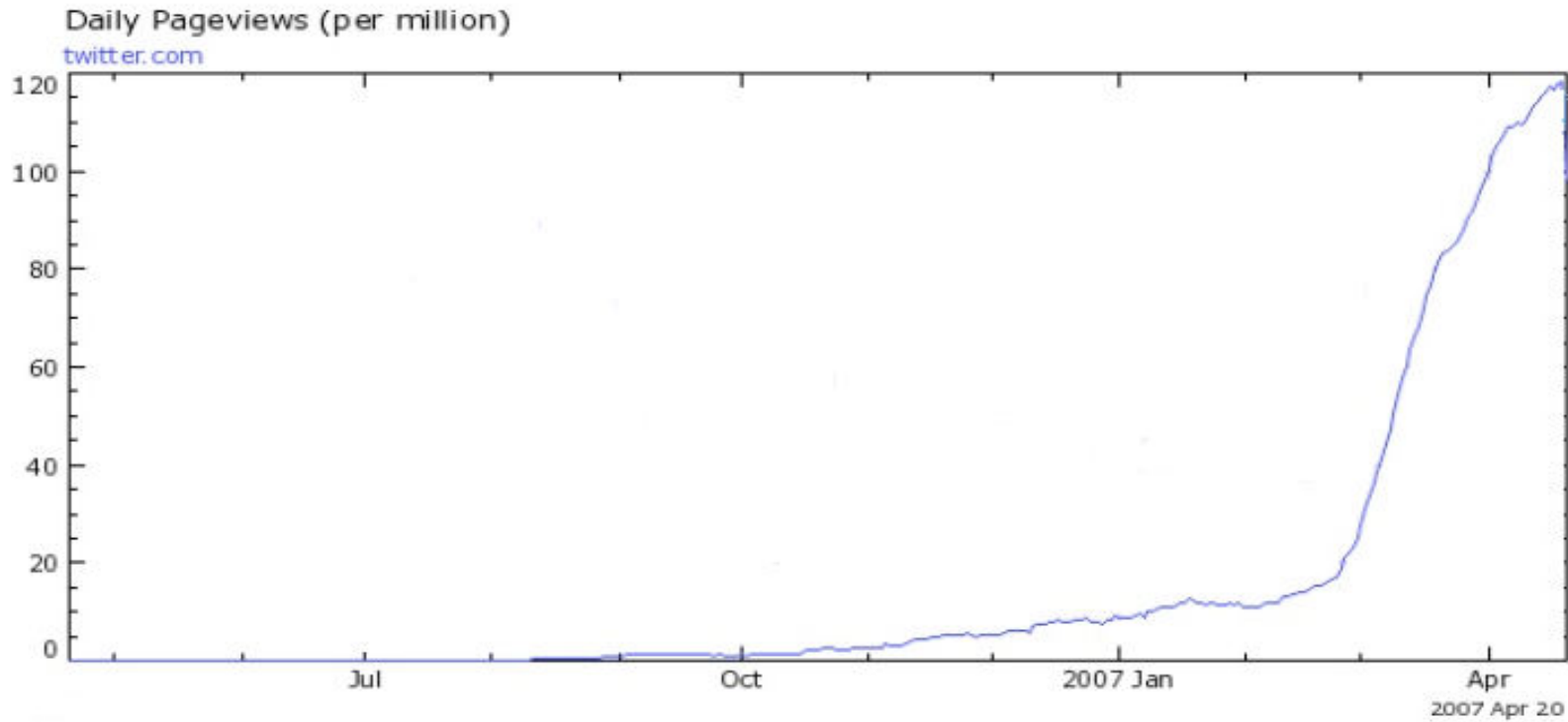


Аналіз отриманих результатів

- ▶ Стандартні налаштування - після 30 000 починається суттєва затримка в обробці даних
- ▶ Після додаткового налаштування - стабільна робота до 90 000 одночасних запитів
- ▶ Ефективність веб-сервера збільшується в 3 рази
- ▶ Проте, що робити, якщо > 90 000 запитів?

Досвід компанії Twitter

- ▶ Популярна соціальна мережа
- ▶ Статистика на початку:
 - ▶ - орієнтовно 350 000 користувачів
 - ▶ - 600 запитів в секунду
 - ▶ - запит оброблюється в середньому 50-100 мс
 - ▶ - серверний парк налічував всього 8 серверів



► Статистика після налаштування:

- - 150 мільйонів активних користувачів по всьому світові
- - 300000 запитів в секунду
- - показники вирости в 500 разів

Що було зроблено?

- ▶ Кешування даних
- ▶ Впровадження черг як основної системи Твіттера
- ▶ Використання CDN для зображень (планується і відео з впровадженням його у сервіс)
- ▶ Повний моніторинг всієї інфраструктури

Підсумки викладеної ідеї

Для збільшення показників продуктивності веб-серверів в реальних навантажених проектах слід переходити від простого налаштування параметрів до модернізації архітектурної складової системи.

Висновки

- ▶ Розглянуті основні вимоги до високонавантаженої системи
- ▶ Розглянуті методи підвищення продуктивності веб-серверів у високонавантажених системах
- ▶ На практиці було проведено тестування налаштованого веб-сервера, аналіз отриманих результатів

Дякую за увагу