

ВИСОКОДОСТУПНИЙ ТА ВІДМОВОСТІЙКИЙ WEB-ДОДАТОК

Вінницький національний технічний університет

Анотація

Розглянуто спосіб реалізації відмовостійкого та високодоступного WEB-додатку на основі програмного забезпечення з відкритим кодом.

Ключові слова: високо доступна та відмовостійка система, Varnish, HAProxy, UCARP, Nginx, Apache, реплікація, MySQL, Open Source, Linux.

Abstract

The method of implementation of a fail tolerant and high availability web application, based on free open source software.

Keywords: high availability and fault tolerant web application, Varnish, HAProxy, UCARP, Nginx, Apache, replication MySQL, Open Source, Linux.

Вступ

Висока доступність - це здатність системи уникати втрати сервісу, мінімізуючи час простою. Це виражається в термінах безперебійності роботи системи, як відсоток від загального часу роботи.

У більшості випадків стратегія безперервності сервісу включатиме як високу доступність, так і відмовостійкість щоб гарантувати, що підтримуються найважливіші функції під час незначних невдач, а також у випадку лиха.

У конкурентному середовищі, сервіс повинен бути постійно доступним і витримувати великі навантаження [6]. У цій публікації проаналізований один із

способів організації такої системи для WEB-додатку.

Постановка задачі

Балансування навантаження є ключовим компонентом високодоступних інфраструктур, які часто використовуються для підвищення продуктивності та надійності WEB-сайтів, програм, баз даних та інших служб шляхом розподілу навантаження на декількох серверах.

WEB-інфраструктура без балансування навантаження може виглядати приблизно так, як показано на рис. 1.

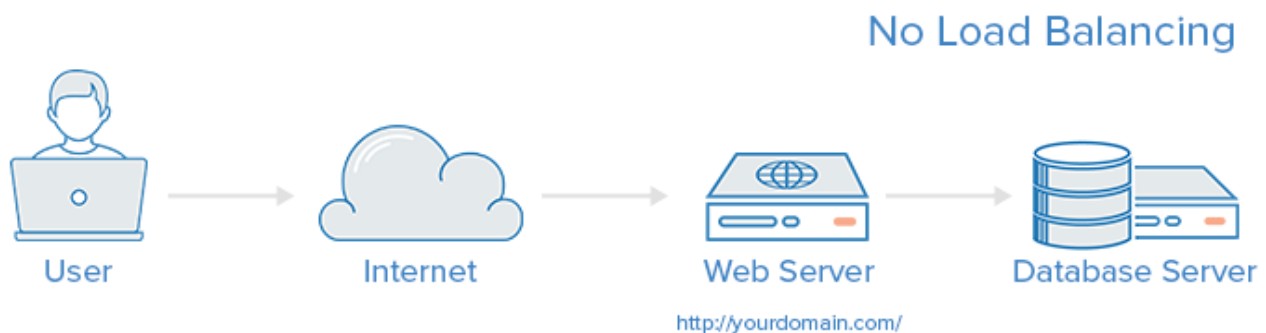


Рис. 1 — WEB-структура без балансування

У цьому прикладі користувач підключається безпосередньо до WEB-сервера на сайті `yourdomain.com`. Якщо цей єдиний WEB-сервер виходить з ладу, користувач більше не зможе отримати доступ до WEB-сайту. Крім того, якщо багато користувачів намагаються одночасно отримати доступ до сервера і ті не в змозі впоратися з завантаженням, користувачі можуть мати повільне завантаження або взагалі не зможуть з'єднатися з WEB-сайтом.

Цю єдину точку збою можна пом'якшити шляхом введення балансування навантаження та, принаймні, одного додаткового WEB-сервера. Як правило, всі сервери надаватимуть однаковий вміст, щоб користувачі отримували відповідний вміст незалежно від того, який сервер їм відповідає.

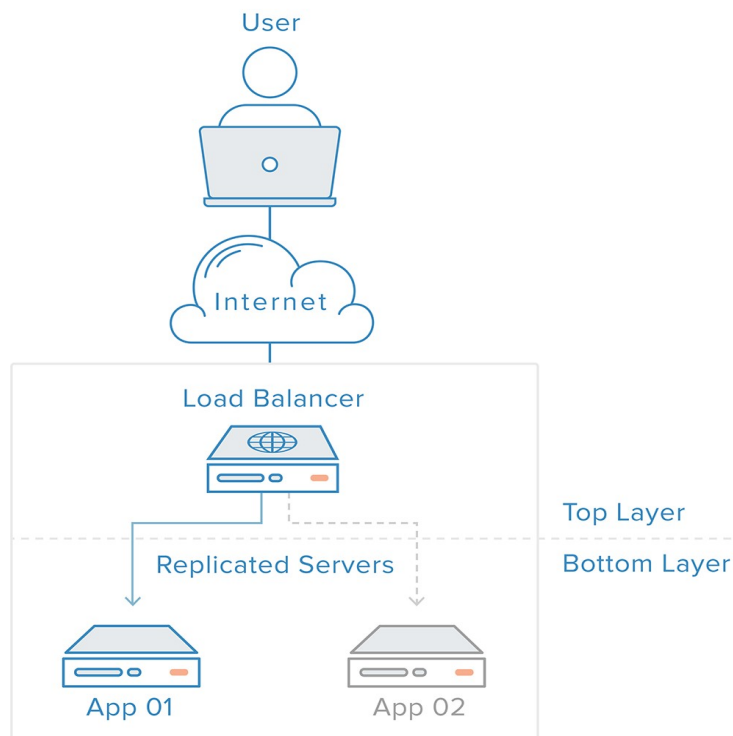


Рис. 2 — WEB-структура з балансуванням

У наведеному вище прикладі, на рис. 2, користувач отримує доступ до балансувальника навантаження, який пересилає запит користувача на сервер, який потім безпосередньо відповідає на запит користувача.

Можливість будь-якої системи зберігати свою працездатність після відмови або виходу з ладу одного або декількох складових компонентів називається відмовостійкістю системи. Відмовостійка система повинна зберігати свою працездатність при виході з ладу мінімум одного вузла, відповідно, основний спосіб підвищення відмовостійкості - створення апаратної надмірності шляхом резервування.

Реалізація

На рис. 3 представлений один із варіантів виконання відмовостійкої та високодоступної системи для WEB-додатку. Було використано створено 2 віртуальні ноди в одному кластері. На кожній ноді знаходиться 3 віртуальні машини: Varnish, WEB-сервер Apache/Nginx, MYSQL. Varnish представляє собою зворотній HTTP проксі-сервер для балансування навантаження. Сервери Varnish та Varnish2 мають одну спільну віртуальну IP адресу (ця адреса вказана у DNS сервері для різних доменів і по ній з сервером зв'язуються користувачі), яка “працює” з одним Varnish сервером. В разі його відмови у дію вступає сервер Varnish2 [1].

Сам Varnish-сервер містить кеш, який використовується для обробки запитів. Завдяки кешуванню зменшується навантаження на WEB-сервер. У разі відсутності необхідної інформації у кеші, надсилається запит до WEB-сервера [3].

Кожний з WEB-серверів (Apache та Nginx) має зв'язок з базою даних через HAProxy, який забезпечує балансування навантаження/високу доступність у разі виходу з ладу одного з серверів баз даних та знаходиться на одній віртуальній машині з WEB-сервером. Якщо впродовж деякого часу сервер баз даних не буде відповідати сервісу HAProxy, то запити почнуть надходити до іншого MySQL-сервера [2]. Це можливо за допомоги реплікації даних “master-master” (бази даних являються рівноправними на відміну від реплікації даних “master-slave”) [4].

Якщо ж потрібно буде налаштувати балансування навантаження на сервери баз даних, то при зміні конфігурації HAProxy, запити зможуть надходити по черзі до БД [5]. У такому випадку, одразу 2 бази будуть працювати з користувачем.

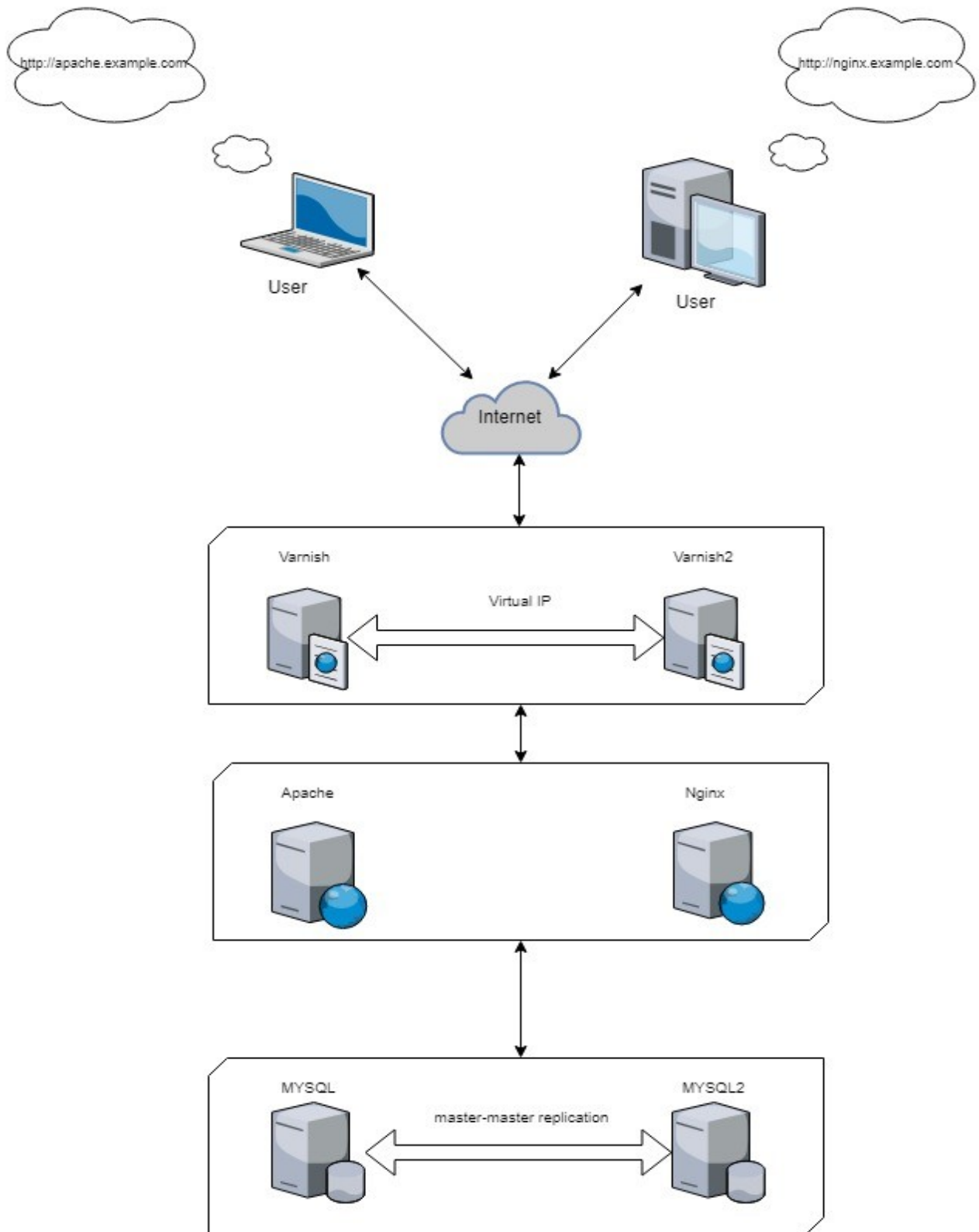


Рис. 3 - Структура відмовостійкої та високодоступної системи

Результати

Варто відзначити, що HTTP-кешування працює тільки з «безпечними» методами, такими як GET, HEAD. Під «безпекою» мається на увазі, що ці методи не будуть змінювати стан даних (редагування, видалення). Ніколи не варто кешувати методи, які змінюють стан (POST, DELETE, PUT, PATCH), так як запити можуть не досягти функціонального рівня виконання на бекенді і будуть оброблені на рівні кеша [7].

За результатами тестування зрозуміло, що Varnish пришвидшує роботу сайту у декілька разів. Незважаючи на те, що Nginx добре відомий своєю швидкістю: середня кількість запитів у секунду була збільшена в три-чотири рази, а час відгуку значно зменшився [8].

Ще одне попередження: Varnish (або, принаймні, версія з відкритим кодом) не підтримує SSL. Отже, якщо вам потрібно Varnish і HTTPS, подумайте про використання іншого проксі-сервера перед ним. Наприклад, haproxy або Varnish з підправленим кодом для роботи з SSL.

Висновки

Переваги даної системи: мінімальний час простою, захист від втрати даних, швидке переключення на резервну копію, висока доступність.

Недоліки: відмовостійкий та високодоступний WEB-додаток може бути дорогим, оскільки така система вимагає безперервної роботи та обслуговування додаткових, надлишкових компонентів.

У даній публікації вказаний лише малий відсоток можливостей високодоступної та відмовостійкої системи. Продуктивність WEB-додатку залежить від: програмного забезпечення, яке на ньому використовується; апаратного забезпечення; правильного налаштування відказостійкості та балансування навантаження.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Плаваюча IP адреса - UCARP [Електронний ресурс] – Режим доступу:
<https://habr.com/post/137476>
2. Проксування серверів - HAProxy [Електронний ресурс] – Режим доступу:
<https://habr.com/post/244027>
3. Кешування даних - Varnish [Електронний ресурс] – Режим доступу:
<https://devdocs.magento.com/guides/v2.1/config-guide/varnish/config-varnish-configure.html>
4. Реплікація Баз даних “master-master” [Електронний ресурс] – Режим доступу: <https://www.linode.com/docs/databases/mysql/configure-master-master-mysql-database-replication>
5. Балансування навантаження [Електронний ресурс] — Режим доступу:
<https://www.digitalocean.com/community/tutorials/what-is-load-balancing>
6. Що таке балансування навантаження? [Електронний ресурс] — Режим доступу: <https://blog.stackpath.com/glossary/load-balancing>
7. Пришвидшення роботи сервера за допомогою Varnish [Електронний ресурс] — Режим доступу: <https://www.sitepoint.com/how-to-boost-your-server-performance-with-varnish>
8. Як пришвидшити роботу сервера за допомогою проксі сервера? [Електронний ресурс] — Режим доступу:
<https://stfalcon.com/ru/blog/post/how-to-speed-site-up-using-varnish-caching-proxy-server>

Уманець Владислав Олександрович — студент групи 1СІ-156, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, Вінниця, e-mail: umanets.vladyslav@gmail.com

Науковий керівник: **Паламарчук Євген Анатолійович** — кандидат технічних наук, доцент кафедри автоматики та інформаційно-вимірювальної техніки, Вінницький національний технічний університет, м. Вінниця

Umanets Vladyslav A. — Department of Computer Systems and Automatic, Vinnytsia National Technical University, Vinnytsia, e-mail : umanets.vladyslav@gmail.com

Supervisor: **Palamarchuk Yevhen A.** — PhD, Docent of Automatics and Informatics and Measurement Techniques Department, Vinnytsia National Technical University, Vinnytsia, email : p@vntu.edu.ua