

АЛГОРИТМ АВТОМАТИЧНОГО МАСШТАБУВАННЯ СЕРВЕРНИХ КОНТЕЙНЕРИЗОВАНИХ РЕСУРСІВ

Вінницький національний технічний університет

Анотація

Запропоновано алгоритм автоматичного масштабування серверних контейнеризованих ресурсів, який дозволяє без участі людини оцінювати навантаження на обчислювальні ресурси в контейнерах та швидко масштабувати їх..

Ключові слова: сервер, контейнер, навантаження, автоматизація, масштабування, деплоймент.

Abstract

The algorithm for an automatic scaling of server's containerized resources, which allows to evaluate the load on computing resources in containers and scale them rapidly without human's participation was proposed.

Keywords: server, container, load, automation, scaling, deployment.

Вступ

Масштабованість – здатність пристрою збільшувати свої можливості шляхом нарощування числа функціональних блоків, виконують одні й ті ж завдання. Іншими словами, у випадку, коли віртуальний сервер не справляється з навантаженням, йому додаватимуться додаткові об'єми потужностей у вигляді додаткових ядер процесора, контейнерів, віртуальних машин, мегабайт оперативної пам'яті, тощо. [1] Автоматичне масштабування виключає з цього процесу участь людини, що значно заощаджує час, ресурси і як наслідок - кошти. [2]

Метою роботи є розробка алгоритму автоматичного масштабування серверних контейнеризованих ресурсів.

Результати дослідження

Запланований програмний продукт відслідковує навантаження різних значущих показників операційної системи і, в залежності від результатів моніторингу, вносить зміни до інфраструктури підвищуючи її стійкість.

Значущими параметрами для моніторингу є наступні параметри:

- CPU
- RAM
- Load Average

Навантаження на CPU (Central Processor Unit - центральний процесор сервера) - це процентне відображення використання ресурсів CPU сервера. [1]

Оперативна пам'ять (англ. Random Access Memory, RAM, пам'ять з довільним доступом) або оперативний запам'ятовуючий пристрій (ОЗП) - енергозалежна частина системи комп'ютерної пам'яті, в якій під час роботи комп'ютера зберігається виконуваний машинний код (програми), а також вхідні, вихідні та проміжні дані, що обробляються процесором.[2]

Середні значення навантаження (Load averages) - це критично важлива для індустрії метрика. Середні значення навантаження в Linux - це «середні значення навантаження системи», що показують потребу в виконуваних потоках (завданнях) у вигляді середньої кількості виконуваних і очікуваних потоків.[1]

Як було сказано вище, алгоритм буде циклічно перевіряти три основні метрики. Задля економії ресурсів перевірка буде виконуватись циклічно один раз на декілька хвилин.

В першу чергу буде запускатись крок дослідження середовища – скільки віртуальних ядер у контейнері, яка кількість оперативної пам'яті. Кількість ядер процесору чисельно дорівнює максимально допустимому значенню Load Average. Також CPU навантаження не має досягати значення 100%. Щодо параметру операційної пам'яті бажано не допускати перевищення реального значення навантаження більше ніж 70-90%. Саме ці значення будуть взяті до уваги при розробці програмного продукту.

Наступним кроком є послідовні циклічні перевірки і порівняння основних значущих параметрів з реальним показниками продуктивності контейнерів. Після кожної перевірки відбуваються зміни в деплоймент файл (якщо показники вийшли за межі встановленої норми; якщо ж ні – то зміни не відбуваються), який зберігається у json форматі. Після збереження змін згідно з деплоймент файлом відбувається відповідне масштабування інфраструктури, яка знаходиться на моніторингу.

Якщо, внаслідок автоматичного масштабування, кількість контейнерів буде перевищувати один, то навантаження буде балансуватись за алгоритмом Round Robin.

Висновки

Встановлено, що запропонований алгоритм дозволяє підвищити стійкість систем побудованих на контейнерах до високих навантажень по різним метрикам, а також завдяки зворотному масштабуванню зекономити кошти.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Критерії оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу: НД ТЗІ 2.2–004–99. – Київ: ДСТСЗІ СБ України, 1999. – 55 с.
2. Марко Беллиньясо Разработка Web-приложений в среде ASP.NET 2.0: задача — проект — решение = ASP.NET 2.0 Website Programming: Problem — Design — Solution. — М.: «Диалектика», 2007. — С. 640. — ISBN 0-7645-8464-2

Гулько Ілля Андрійович — студент групи УБ-18м, факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, Вінниця, e-mail: illia.gunko69@gmail.com

Науковий керівник: *Карпинець Василь Васильович* — к.т.н, доцент, Вінницький національний технічний університет, м. Вінниця

Gunko Illia A. — Department of Management and Informational Security, Vinnytsia National Technical University, Vinnytsia, email: illia.gunko69@gmail.com

Supervisor: *Karpinets Vasyl V.* — PhD, Docent, Vinnytsia National Technical University, Vinnytsia