

Zdecentralizowany system do zarządzania aplikacjami chmurowymi w postaci mikrousług działających wewnątrz kontenerów wirtualizacyjnych

Grzegorz Dwornicki

Architektura mikrousług jest to technika projektowania skalowalnych oraz odpornych na awarię aplikacji chmurowych, która przenosi złożoność tradycyjnej aplikacji monolitycznej na infrastrukturę. Każda z mikrousług odpowiada tylko za jeden funkcyjny wymóg aplikacji i jest zaimplementowana w osobnym kontenerze wirtualizacyjnym, który może być uruchomiony, skalowany i przenoszony niezależnie od innych przy użyciu automatycznych systemów zarządzania chmurą.

Napisana zgodnie z architekturą mikrousług aplikacja może działać szybciej, jeśli komunikujące się ze sobą często mikrousługi znajdują się na tym samym serwerze. Dlatego też stosowane obecnie scentralizowane systemy zarządzające mikrousługami zaimplementowanymi wewnątrz kontenerów wirtualizacyjnych, takie jak Docker Swarm, Kubernetes, czy też Mesos starają się optymalizować ich rozmieszczenie na serwerach chmury przy uruchamianiu i potem podczas pracy aplikacji. Ich działanie opiera się na założeniu, że przeniesienie kontenera na inny serwer polega na jego zatrzymaniu, przeniesieniu obrazu i ponownym uruchomieniu na serwerze docelowym. Czas niedziałania kontenera jest wówczas taki sam, jak czas jego przenoszenia pomiędzy serwerami.

Założenie to nie jest spełnione podczas zastosowania migracji kontenerów na żywo. Podczas przenoszenia kontener nadal działa, znajduje się fizycznie jednocześnie w dwóch miejscach i traci kontakt z siecią na czas 10 razy krótszy od czasu przenoszenia. Dlatego też potrzebny jest zdecentralizowany system zarządzający kontenerami. System taki pozwoliłby optymalizować rozłożenie kontenerów na serwerach chmury przy mniejszych przerwach w pracy mikrousług. Jego zastosowanie powinno się zatem przełożyć na mierzalnie wyższą wydajność aplikacji chmurowych.

Proponowany zdecentralizowany system zarządzający mikrousługami działającymi w kontenerach OpenVZ zainspirowany zachowaniem rojów autonomicznych robotów z oddziaływaniem feromonowym. Odpowiednikiem ruchu robota jest migracja kontenera na żywo. Każdy kontener sam podejmuje decyzje czy migrować oraz gdzie migrować. Aby kontener mógł sam zlecić migrację musi mieć możliwość zebrania potrzebnych informacji na podjęcie takiej decyzji. Służą do tego feromony przyciągające lub odpychające kontenery, które wyliczane są niezależnie przez każdy serwer na podstawie wymagań i liczby obecnych na nim kontenerów oraz zasobów serwera.

Prezentowany system zarządzający kontenerami wirtualizacyjnymi na serwerach chmury obliczeniowej nie ma centrum zarządzającego i dlatego też jest lepiej skalowalny i odporny na awarię od istniejących systemów. Jest to kolejna oprócz zwiększenia wydajności aplikacji korzyść z jego potencjalnego zastosowania w praktyce.