mogą być też dodatkowe kontrole przeprowadzane przez jądro. Możliwe jest także wykorzystanie hybrydowych rozwiązań, łączących większą izolację maszyn wirtualnych lub oddzielne systemy fizyczne z łatwością wdrażania kontenerów.

Jeśli kontenery zawierają pełną kopię systemu operacyjnego i pozwalają administratorom na logowanie, to stanowią w zasadzie miniaturowe maszyny wirtualne. Mimo że kontenerów można używać w trybie „minimaszyny wirtualnej", nie jest to najlepszy sposób ich wykorzystania. Strategia zarządzania zasobami dla kontenerów zależy częściowo od tego, w jaki sposób są wykorzystywane. Poniżej omówiono dwa modele: „natywny" model kontenera oraz model „minimaszyny wirtualnej".

**Natywny model kontenera.** W natywnym modelu kontenera:

- Kontenery powinny zawierać absolutnie minimalne elementy systemu operacyjnego niezbędne do wykonywania ich funkcji.
- Każdy kontener powinien spełniać tylko jedną funkcję (lub „sprawy" w pewnej dokumentacji).
- Kontenery powinny być niezmienne, co oznacza, że nie powinny zmieniać się z czasem. Kontenery mogą wprowadzać zmiany w niektórych innych komponentach, takich jak zapis danych do usługi magazynowania, ale pamięć ta powinna być utrzymywana niezależnie od samego kontenera.
- Niezmienne kontenery powinny pozostawać doskonałą kopią kodu zawartego w obrazie przez cały okres ich życia. Kod nie powinien być aktualizowany przez samego siebie ani żadną inną logującą się do niego osobę. Zamiast aktualizacji kontenerów, stare kontenery powinny być kasowane, a na ich miejsce tworzone nowe kontenery ze zaktualizowanym kodem.

W przypadku natywnych, niezmiennych kontenerów nie powinien istnieć żaden wymóg rutynowej konserwacji wykonywanej przez administratorów logujących się do kontenerów, aczkolwiek należy zapewnić pewne, awaryjne możliwości uzyskania dostępu. Jeśli logowanie do kontenerów nie jest ogólnie dozwolone, zarządzanie dostępem do kontenerów staje się mniej ryzykowne niż w przypadku serwerów. Zarządzanie podatnościami i konfiguracją pozostają nadal istotnymi zagrożeniami, aczkolwiek zakres dla danego kontenera jest znacznie węższy niż zakres dla serwera, który na ogół może wykonywać wiele różnych funkcji.

Natywne kontenery są generalnie tworzone i kasowane znacznie częściej niż maszyny wirtualne. Oznacza to, że bardziej sensowne jest inwentaryzowanie obrazów kontenerów niż samych kontenerów i następnie śledzenie, z którego obrazu kopiowany jest kontener. Obraz kontenera musi zostać zinwentaryzowany przede wszystkim w celu śledzenia oprogramowania i konfiguracji obrazu, tak aby obraz mógł zostać zaktualizowany o poprawki bezpieczeństwa oraz nowe konfiguracje w miarę wykrycia podatności na zagrożenia.

**Model kontenera „Minimaszyna wirtualna".** W modelu, w którym kontenery są traktowane jak miniaturowe maszyny wirtualne:

- Kontenery zwykle uruchamiają pełną kopię komponentów trybu użytkownika systemu operacyjnego.
- Kontenery pełnią wiele funkcji, takich jak uruchamianie dwóch różnych rodzajów usług w tym samym kontenerze.
- Kontenery umożliwiają logowanie administracyjne i zmieniają się z czasem.

Jeśli kontenery są wykorzystywane jako minimaszyny wirtualne, powinny być inwentaryzowane i chronione w taki sam sposób jak maszyny wirtualne. Oznacza to instalowanie agentów do inwentaryzacji oraz śledzenie użytkowników, oprogramowania i wszystkich innych elementów wymienionych w poprzedniej części dotyczącej maszyn wirtualnych.

W obu modelach obrazy powinny być inwentaryzowane i aktualizowane tak, aby nowo tworzone kontenery były pozbawione podatności na zagrożenia.

**Systemy orkiestracji kontenerów.** Kontenery są świetne, ale jeszcze lepiej jest mieć coś, co może być wykorzystane do: łączenia kontenerów w pakiety w celu wykonywania funkcji wyższego poziomu, uruchamiania wielu kopii tych pakietów, równoważania obciążenia tych kopii i udostępniania innych funkcjonalności, takich jak łatwe sposoby komunikowania się komponentów między sobą. Taki typ systemu nazywany jest systemem orkiestracji kontenerów.

Najpopularniejszą implementacją orkiestracji kontenerów jest Kubernetes z kontenerami Docker. We wdrożeniu Kubernetes podstawowymi zasobami są klastry, w których przechowywane są pods (zasobniki), w których z kolei przechowywane są kontenery Docker kopiowane z obrazów. W środowisku Kubernetes należy rozważyć inwentaryzację następujących komponentów:

- Klastry Kubernetes, dzięki czemu można kontrolować dostęp do nich, a oprogramowanie Kubernetes może być aktualizowane. Podatności na zagrożenia w oprogramowaniu Kubernetes mogą stanowić zagrożenia dla wszystkich działających w nich zasobnikach.
- Kubernetes pods, które mogą zawierać jeden lub więcej kontenerów Docker. Za pomocą wiersza poleceń Kubernetes lub interfejsu API można śledzić istniejące pods oraz kontenery, które tworzą te pods.
- Obrazy kontenerów Docker.

### Aplikacja Platforma jako usługa

Oferty aplikacji działających na zasadzie Platforma jako usługa (aPaaS), takie jak Cloud Foundry lub AWS Elastic Beanstalk, umożliwiają wdrożenie kodu bez samodzielnego przydzielania maszyn wirtualnych. Oferty te obejmują również wiele innych zasobów,