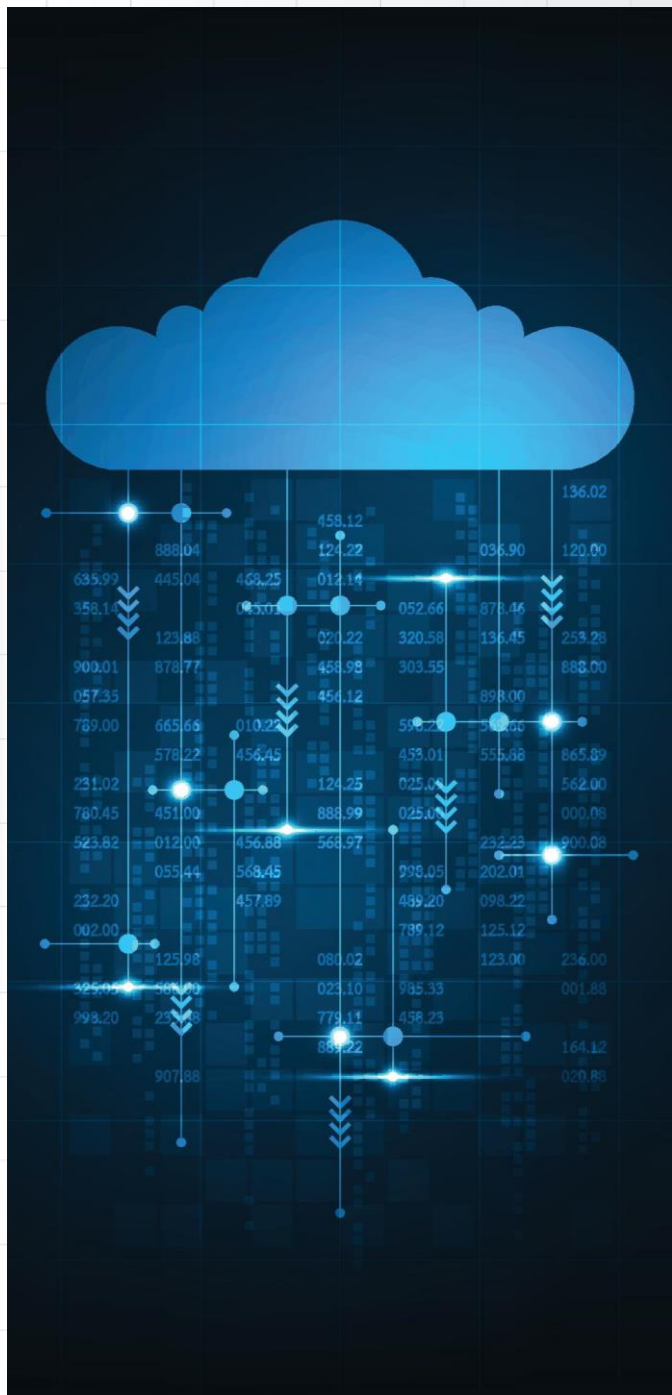




Wrocław
University
of Science
and Technology



Programowanie w chmurze

Rafał Palak

Politechnika Wrocławska



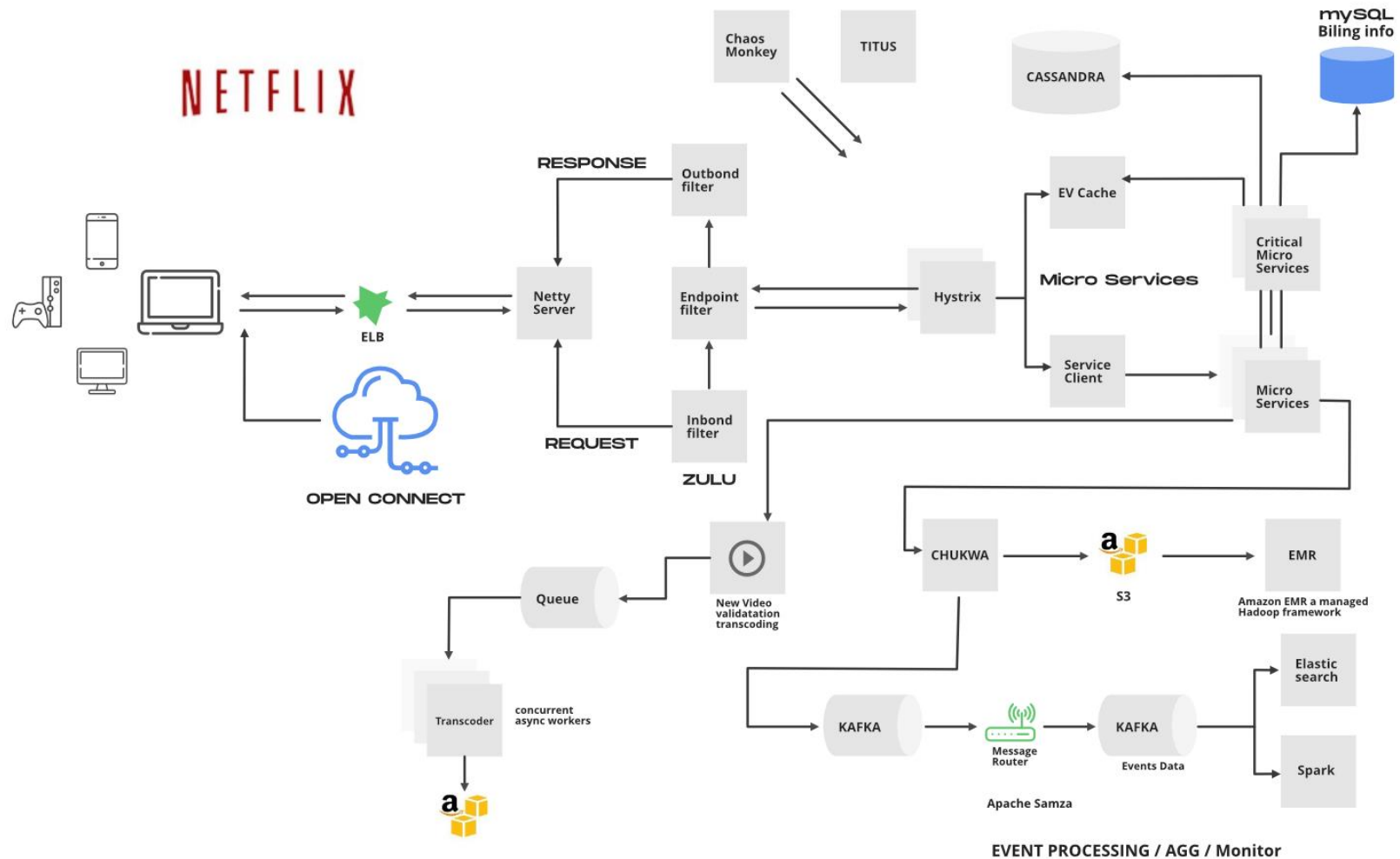
Wrocław
University
of Science
and Technology

Najistotniejsze zagadnienia

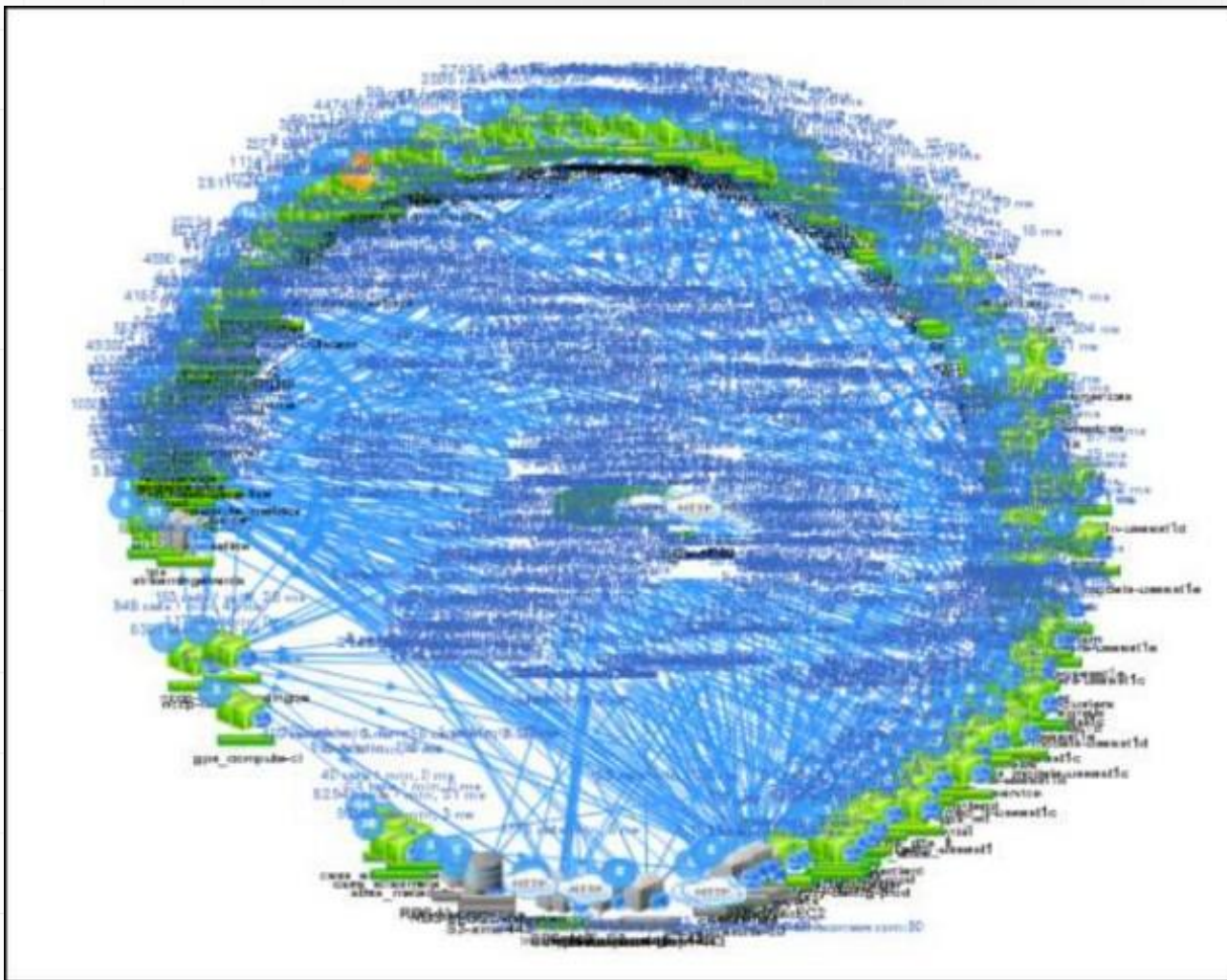
Chmura obliczeniowa (Cloud computing) – dlaczego?



Chmura obliczeniowa (Cloud computing) – skalowalność

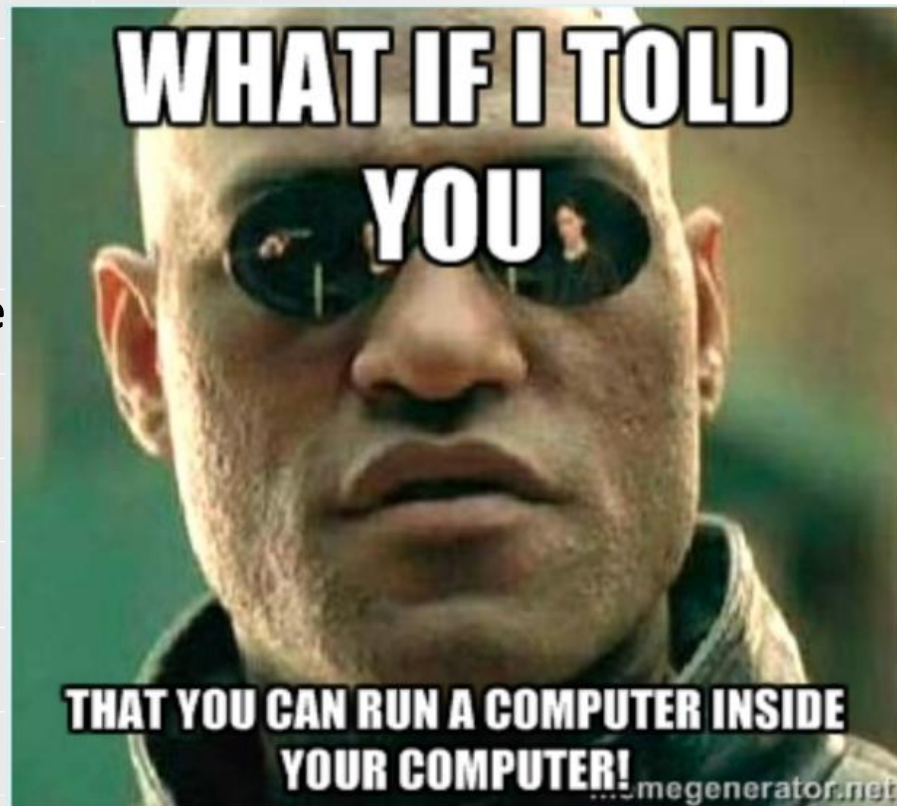


Chmura obliczeniowa (Cloud computing) – skalowalność



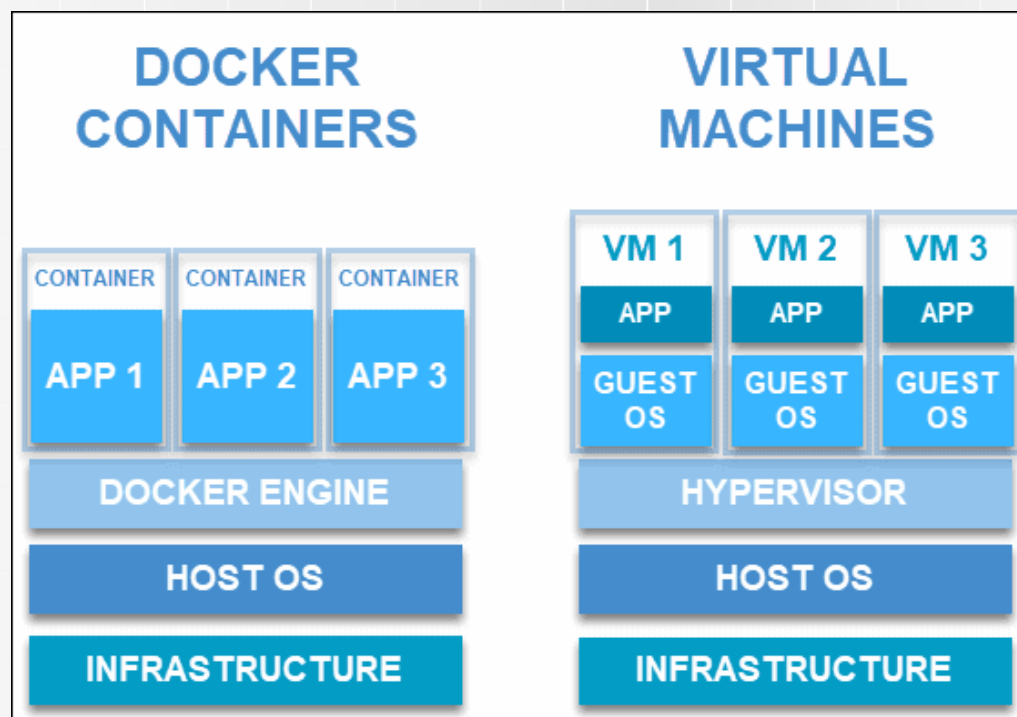
Docker

- **Platforma oprogramowania** służąca do **automatyzacji procesu wdrażania, skalowania oraz zarządzania aplikacjami przez konteneryzację**. Pozwala deweloperom „**pakować**” aplikacje **wraz z ich zależnościami** w standardowe jednostki oprogramowania, zwane **kontenerami**, które są **izolowane od środowiska i jednolicie działają na różnych systemach i infrastrukturach**.



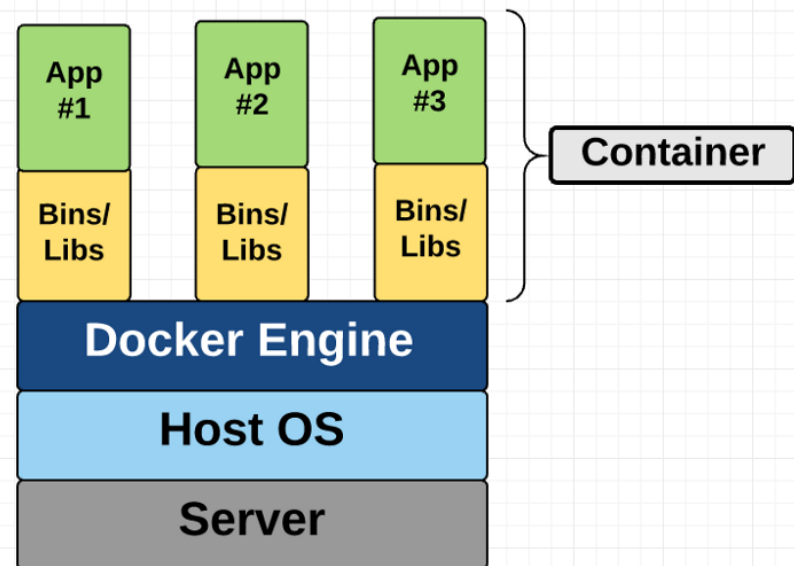
Maszyna wirtualna

- Emulowany komputer z **własnym systemem operacyjnym**
- Uruchomiony na oprogramowaniu zwanym **hypervisorem**, działającym na **fizycznym sprzęcie komputerowym**
- Pozwala na uruchomienie wielu systemów operacyjnych na **jednym fizycznym serwerze**, każdy w izolowanym środowisku.



Kontener [1]

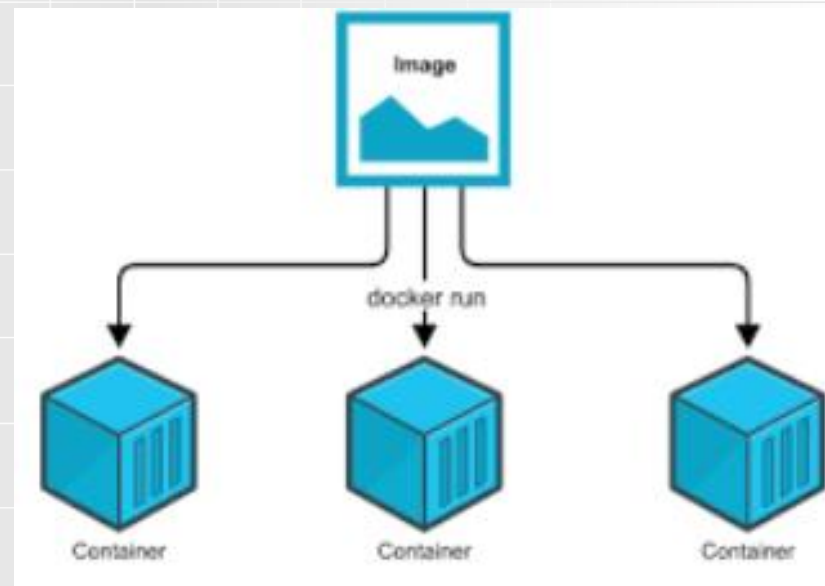
- Kontener to **proces uruchomiony w izolowanym środowisku na maszynie gospodarza, który jest odizolowany od wszystkich innych procesów działających na tej maszynie gospodarza.**



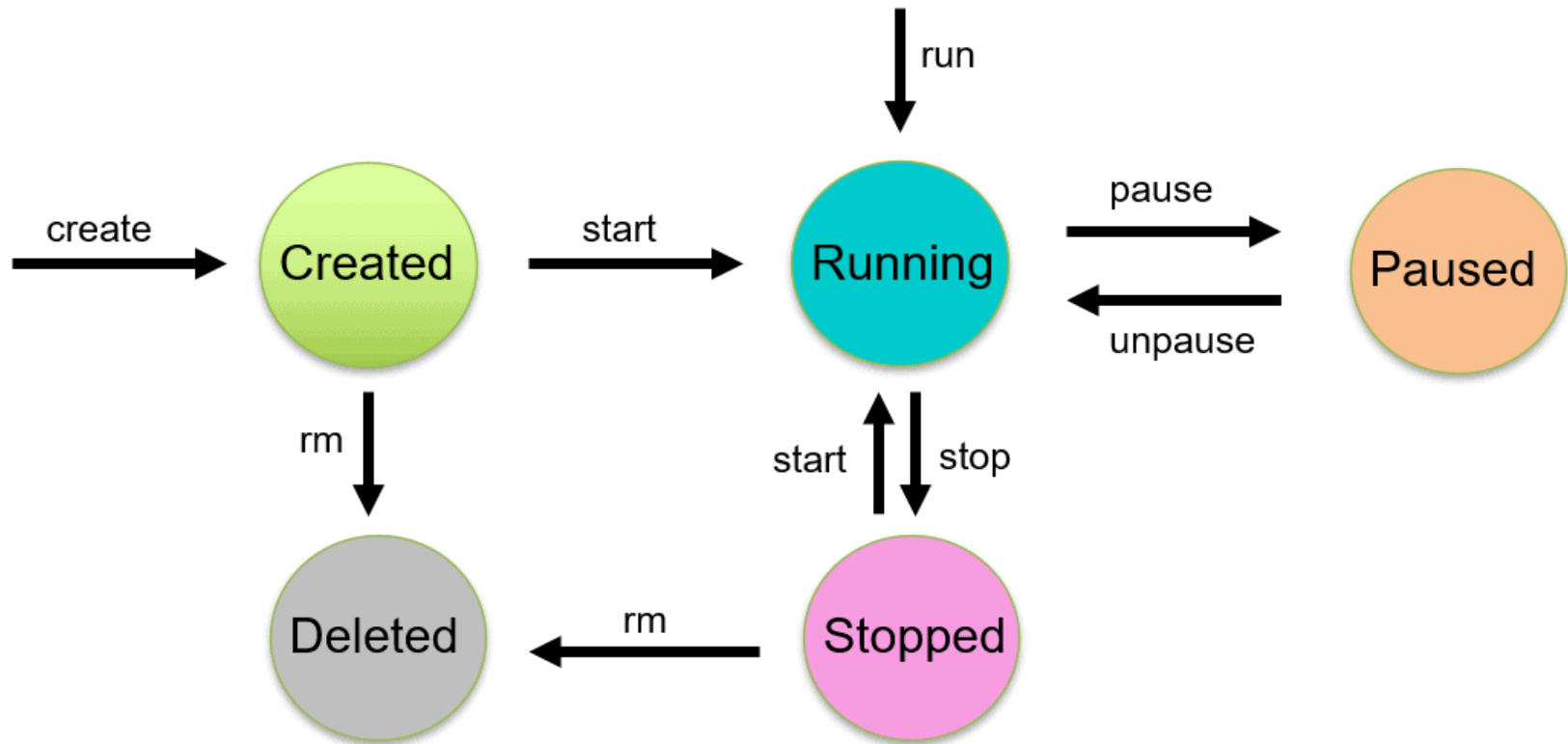
<https://docs.docker.com/get-started/>

Obraz

- Działający kontener korzysta z izolowanego systemu plików. **Ten izolowany system plików jest dostarczany przez obraz**, który musi zawierać wszystko, co potrzebne do uruchomienia aplikacji - wszystkie zależności, konfiguracje, skrypty, pliki binarne itp. Obraz zawiera także inne konfiguracje kontenera, takie jak zmienne środowiskowe, polecenie domyślne do uruchomienia i inne metadane.

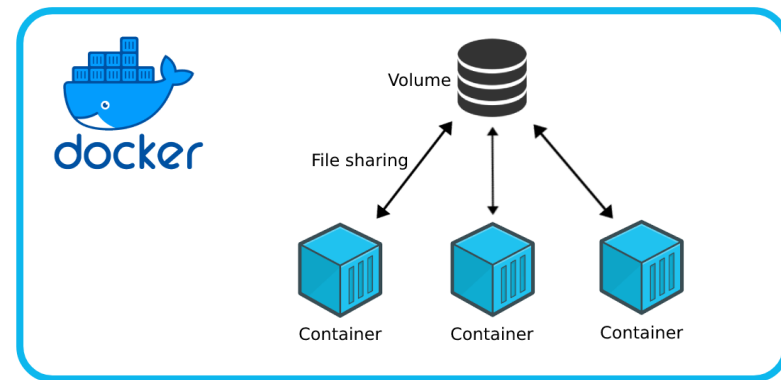


Cykl życia obrazów Docker



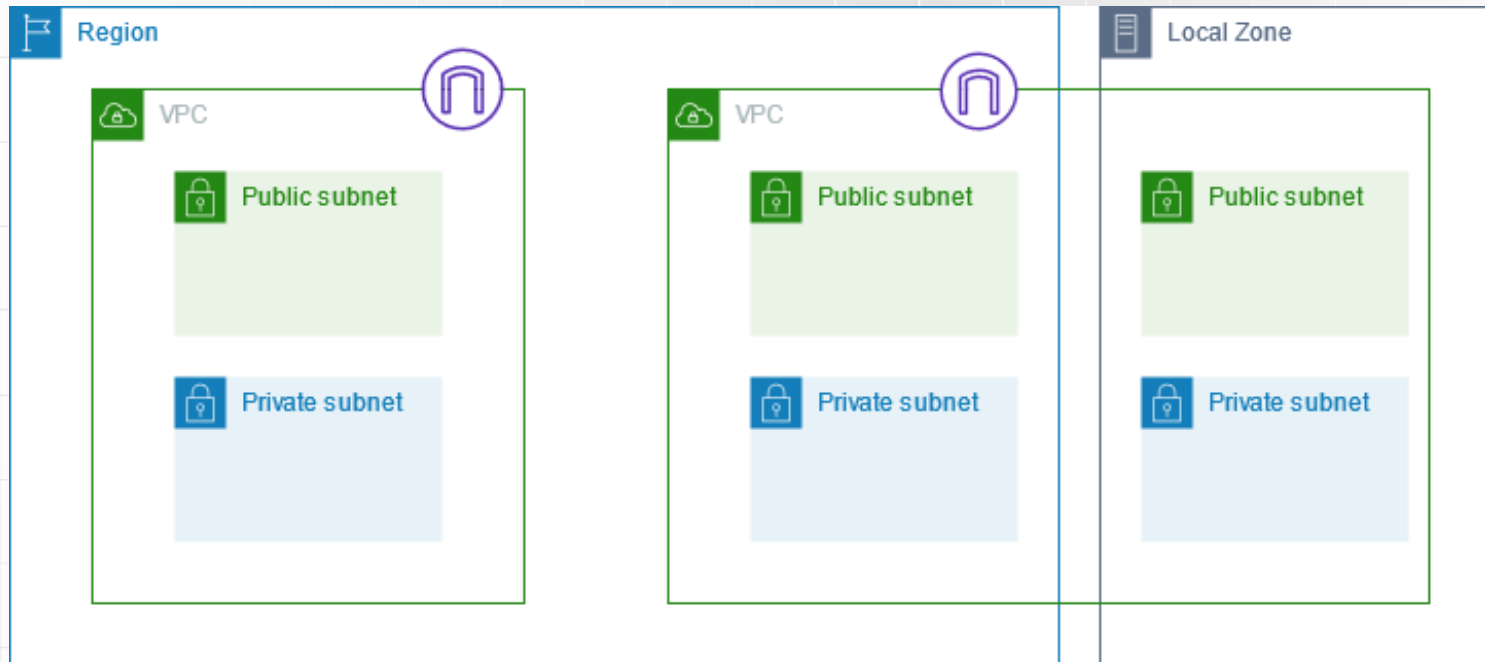
Wolumeny

- `docker volume create` – tworzy nowy wolumen
- `docker volume ls` – pozwala przeglądać listę wolumenów
- `docker volume inspect` – wyświetla szczegółowe informacje o konkretnym wolumenie.
- `docker volume rm` – usuwa wolumen
- `docker run -v moj_wolumen:/app/data moj_obraz`
- To mechanizm trwałego przechowywania danych, używany do zarządzania danymi, które powinny przetrwać usunięcie lub restart kontenerów
- Mogą być bezpiecznie udostępniane między wieloma kontenerami



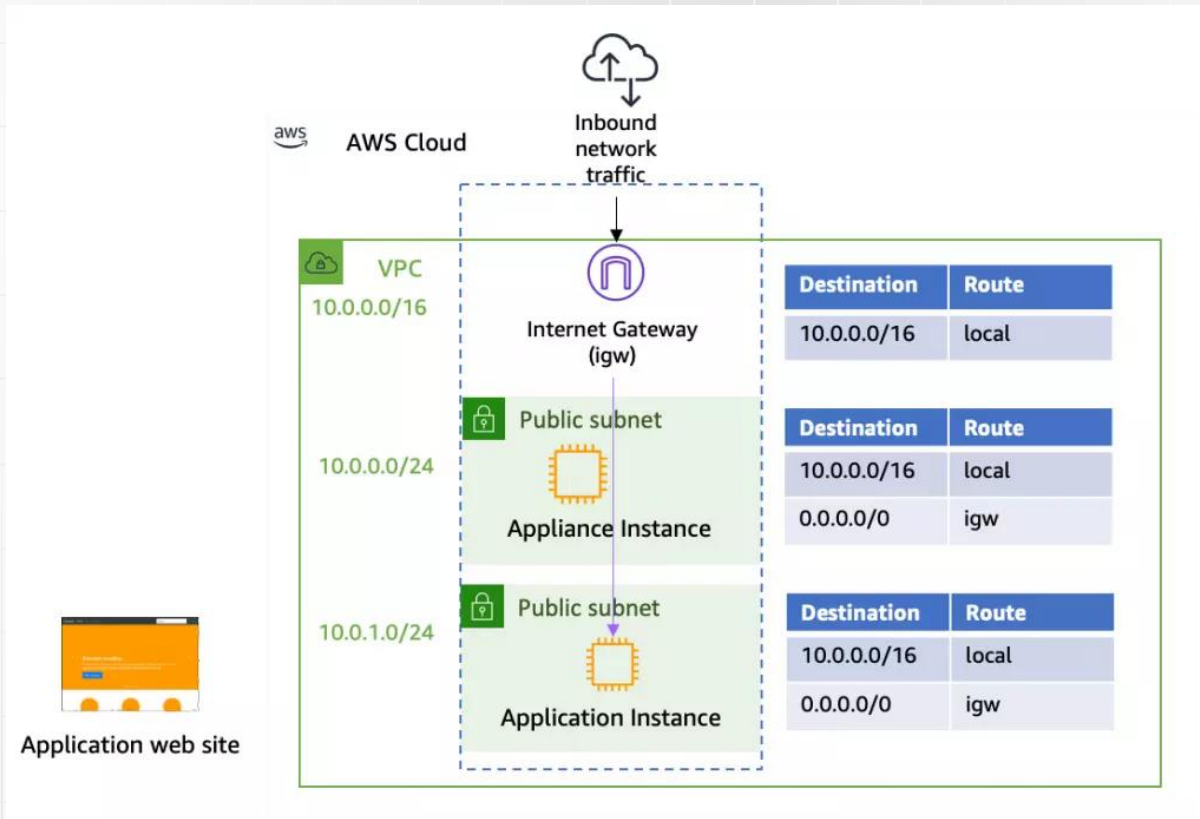
Podsieci (Subnets)

- Pozwalają na podział VPC na mniejsze, izolowane sekcje. Można tworzyć podsieci publiczne i prywatne.



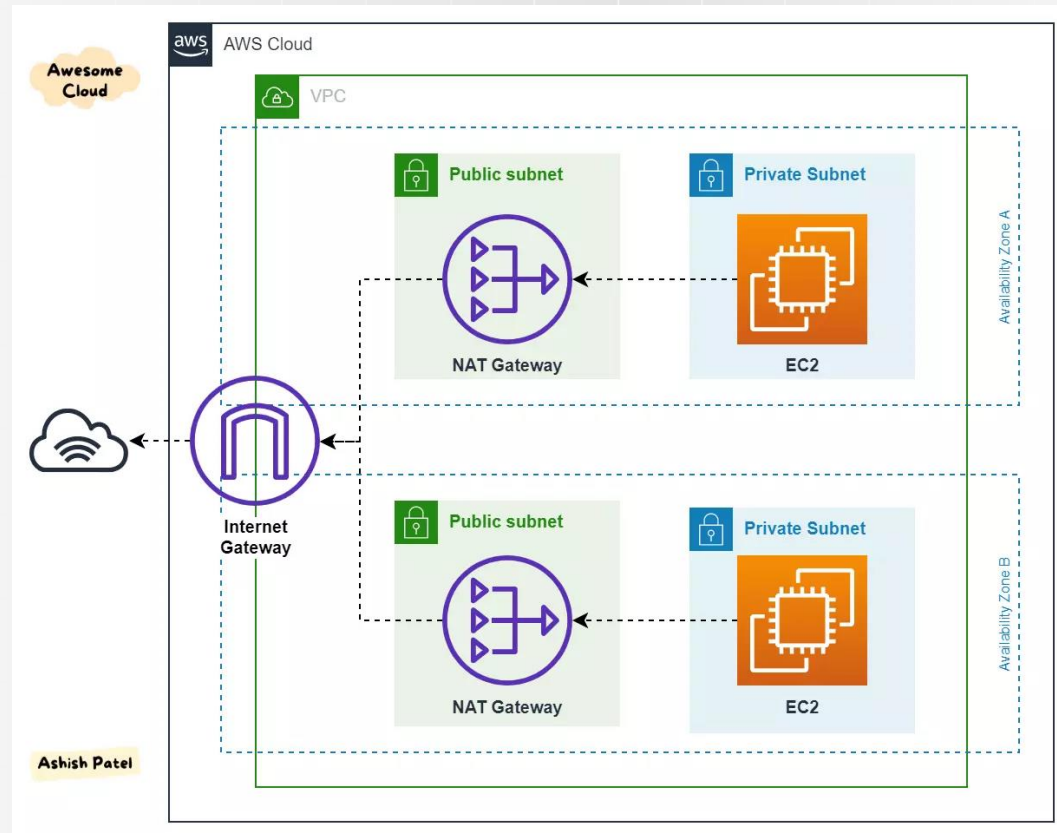
Tablice tras (Route Tables)

- To zestawy reguł (tras), które określają, jak ruch sieciowy jest kierowany w wirtualnej sieci VPC.



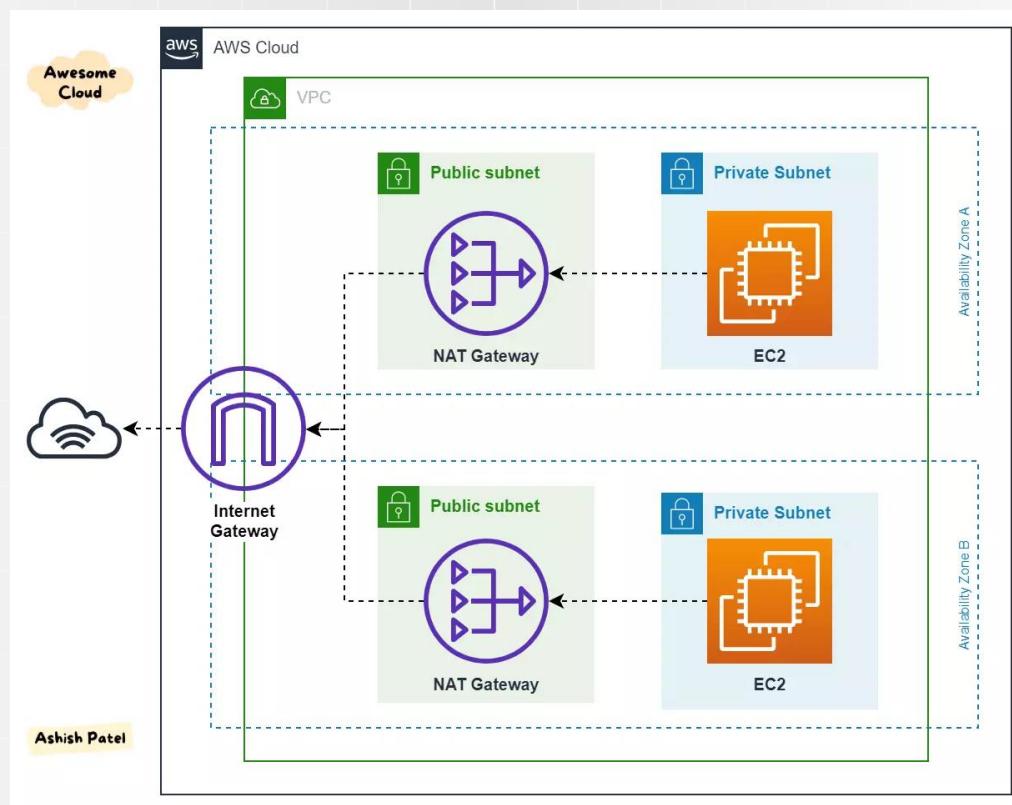
Internet Gateway

- Komponent VPC, który umożliwia komunikację między zasobami w VPC a Internetem.



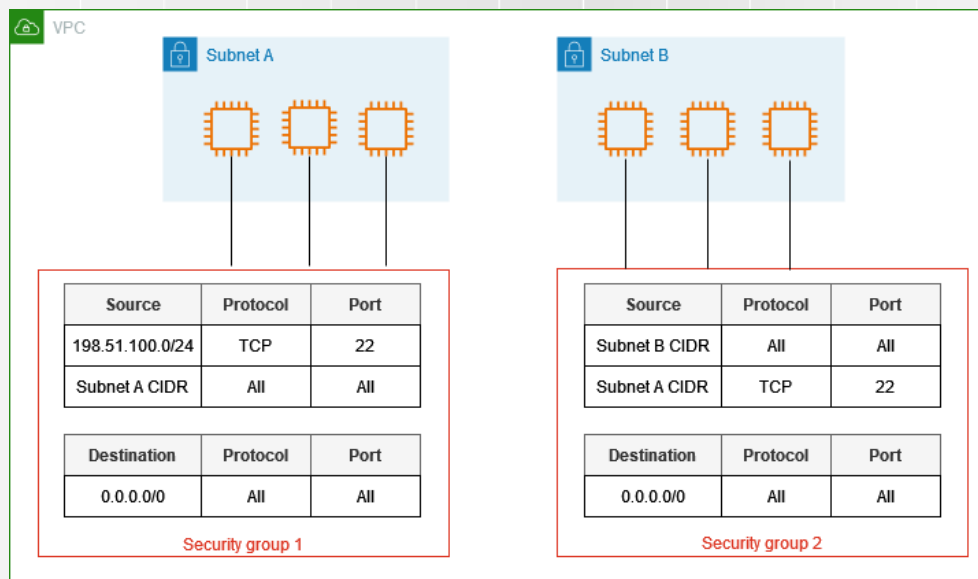
NAT Gateway

- Jest kluczowym elementem infrastruktury sieciowej w Amazon Web Services (AWS). Służy do umożliwienia instancjom w prywatnych subnetach (podsięciach) dostępu do Internetu lub innych usług AWS bez ujawniania ich prywatnych adresów IP.



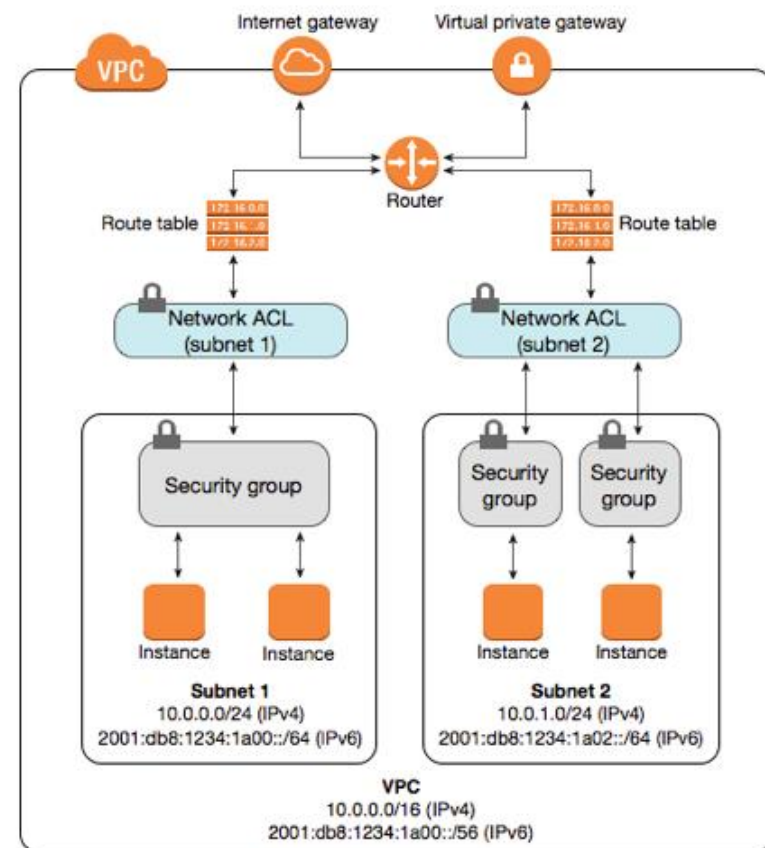
Security Groups

- To wirtualne zapory ogniowe (firewalle), które kontrolują ruch przychodzący i wychodzący do zasobów uruchamianych w Amazon VPC. Są to zestawy reguł, które definiują, jaki ruch jest dozwolony do i z instancji EC2 oraz innych zasobów w ramach VPC.



Network ACLs (Access Control Lists)

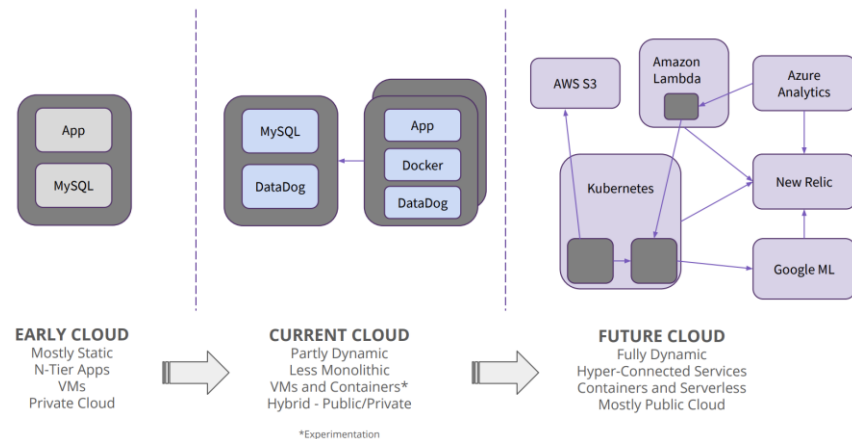
- To jedna z warstw zabezpieczeń w Amazon VPC, które pełnią ważną rolę w zarządzaniu ruchem sieciowym do i z podsieci. Główne cele:
 - Zarządzanie ruchem sieciowym: Network ACLs kontrolują ruch przychodzący i wychodzący na poziomie podsieci, umożliwiając określenie, które ruchy są dozwolone, a które zablockowane.
 - Dodatkowa warstwa zabezpieczeń: Działają jako dodatkowa warstwa zabezpieczeń obok Security Groups, które działają na poziomie instancji.



Infrastruktura jako kod (IaC) [1]

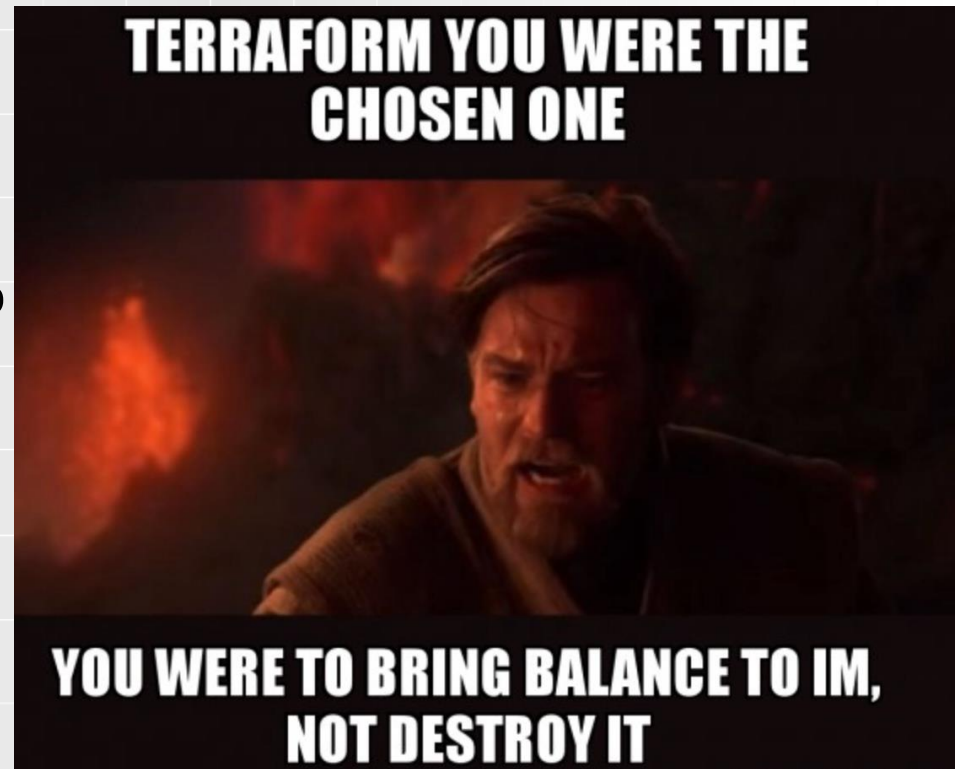
- Zarządzanie i udostępnianie infrastruktury za pomocą kodu, a nie procesów ręcznych
- Tworzone są pliki konfiguracyjne zawierające specyfikacje infrastruktury, co ułatwia edycję i dystrybucję konfiguracji
- Za każdym razem tworzone jest to samo środowisko
- Kodyfikując i dokumentując specyfikacje konfiguracji, IaC wspomaga zarządzanie konfiguracją i pomaga uniknąć nieudokumentowanych, doraźnych zmian konfiguracyjnych

Cloud Transition



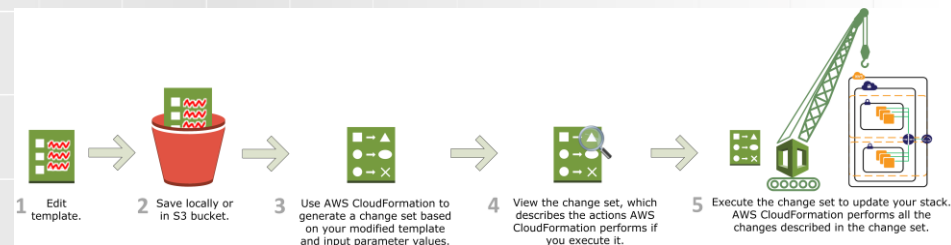
Terraform

- Jedno z popularniejszych **narzędzi do zarządzania infrastrukturą jako kod** (Infrastructure as Code, IaC)
- Pozwala **opisać strukturę swojej infrastruktury** za pomocą języka konfiguracyjnego **HashiCorp Configuration Language (HCL)** lub **JSON**
- **Projekt** o niesamowitej elastyczności, **obsługujący wszystkie najpopularniejsze platformy chmurowe**



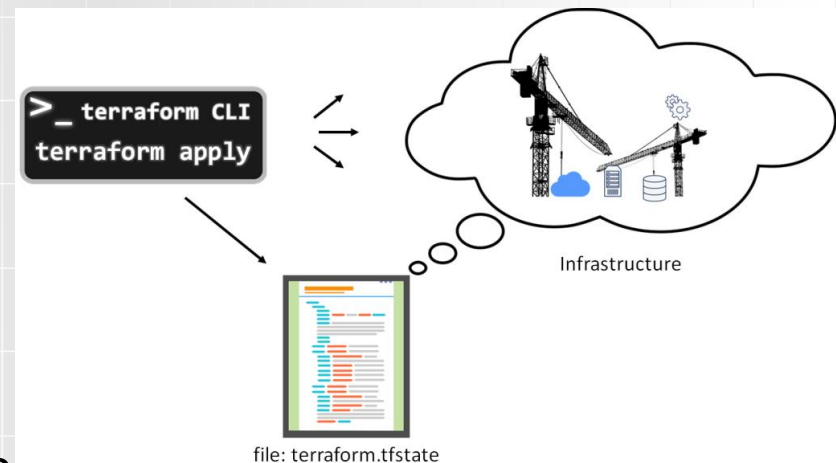
CloudFormation

- Działa tylko dla AWS
- Jest w **pełni zintegrowany z AWS** pozwalając na większą kontrolę
- Pozwala opisać strukturę swojej infrastruktury za pomocą **YAML** and **JSON**
- **Oferuje rollback** pozwalający przywrócenie aplikacji do poprzedniego stanu
- **Ścisły związek tego narzędzia z AWS** umożliwia wdrażanie infrastruktury w kilku regionach i kontach przy użyciu tego samego szablonu CloudFormation.
- Odpowiedniki u konkurencji:
Google Cloud Deployment Manager oraz **Azure Resource Manager**



Stan [1]

- Odnosi się do **aktualnej konfiguracji i właściwości zasobów infrastruktury**, które są zarządzane przez Teraform
- **Plik stanu jest używany do mapowania zasobów w konfiguracji Teraforma do rzeczywistych zasobów w usłudze chmurowej**
- Pozwala Teraformowi na **śledzenie informacji o zarządzanych zasobach**, co jest kluczowe dla odpowiedniego **planowania i wprowadzania zmian w infrastrukturze**
- Pozwala Teraform **wiedzieć, które zasoby istnieją i jakie są ich aktualne właściwości**



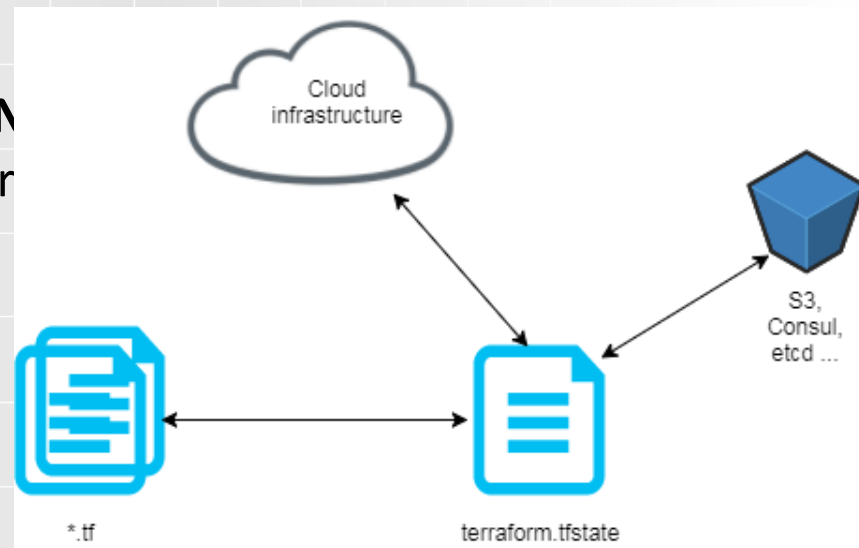
Stan [2]

- Kluczowy dla **uniknięcia konfliktów i niespójności**, gdy różne osoby lub systemy wprowadzają zmiany w infrastrukturze
- Pozwala Teraformowi **optymalizować operacje**, tylko **wprowadzając zmiany tam, gdzie są one naprawdę potrzebne**, zamiast rekreować wszystko od zera za każdym razem
- Używany do **wyświetlania wartości wyjściowych** (outputs) po zastosowaniu zmian, co **pozwala na łatwe przekazywanie informacji między różnymi modułami i konfiguracjami** Teraforma.



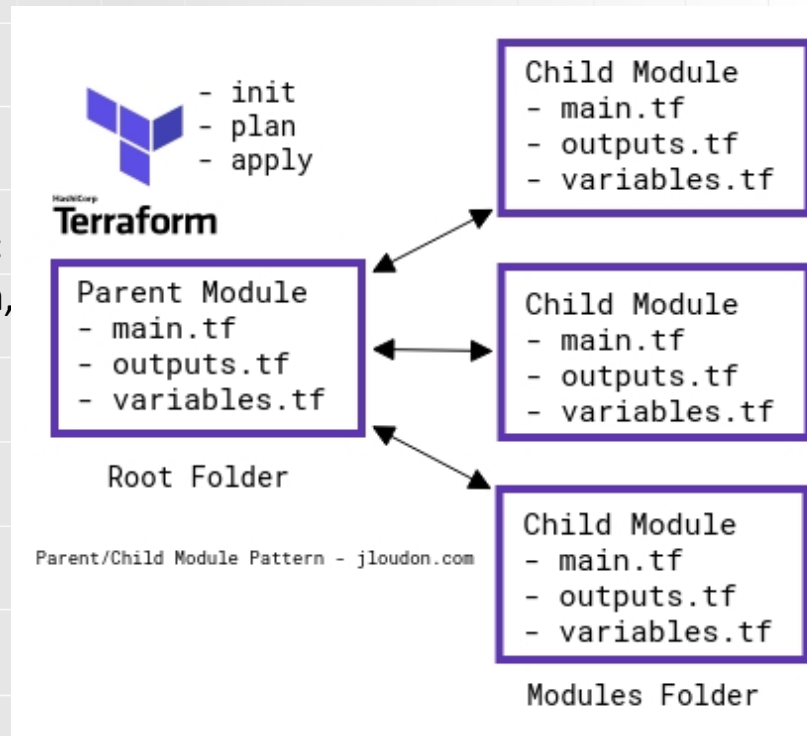
Stan [3]

- Teraform umożliwia zarządzanie stanem lokalnie lub zdalnie
- Przechowywanie **stanu w sposób zdalny** jest zalecane dla **większych zespołów i infrastruktur**, ponieważ zapewnia lepszą kontrolę i bezpieczeństwo
- **Plik stanu** jest zazwyczaj **plikiem JSON** który można przeglądać i zarządzać przy użyciu różnych poleceń Teraform, takich jak **terraform state list** czy **terraform state show**.



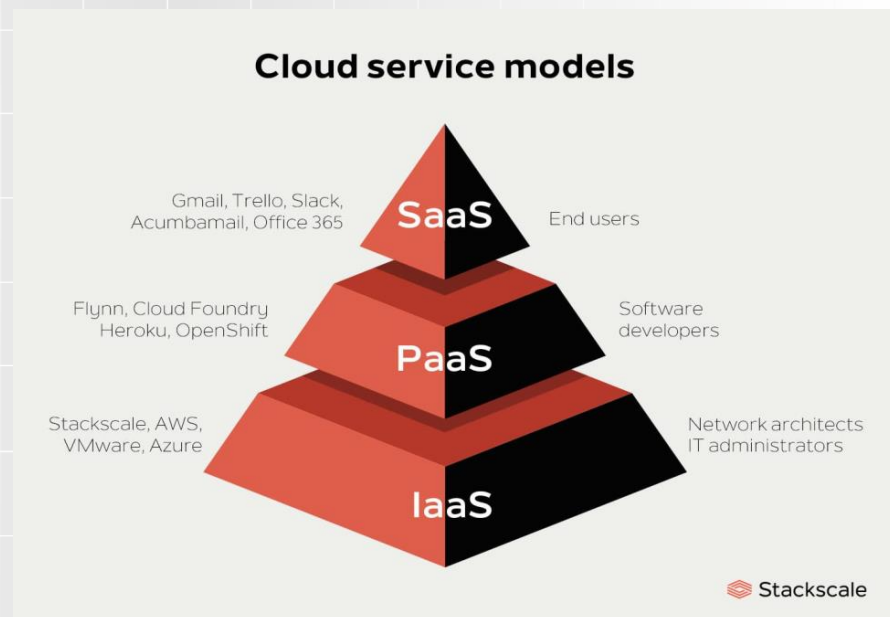
Moduły – jak używać

- **Definiowanie Modułu** - Moduł jest definiowany poprzez utworzenie nowego katalogu i umieszczenie w nim plików konfiguracyjnych Teraforma. Każdy moduł **powinien mieć swoje własne pliki konfiguracyjne**, które definiują zasoby, które moduł będzie zarządzać.
- **Użycie Modułu** - Aby użyć modułu, należy dodać blok module do głównej konfiguracji Teraforma, określając ścieżkę do katalogu modułu oraz ewentualne wejścia (inputy), które moduł wymaga.
- **Parametry Wejściowe i Wyjściowe** - Moduły mogą przyjmować parametry wejściowe (zmienne), które pozwalają na konfigurowanie ich działania. Moduły mogą również zwracać wartości wyjściowe, które mogą być używane przez inne części twojej konfiguracji Teraforma.
- **Moduły Publiczne i Prywatne** - Można tworzyć własne moduły lub korzystać z modułów dostępnych publicznie w Rejestrze Modułów Teraforma, gdzie społeczność udostępnia moduły dla różnych dostawców i usług.



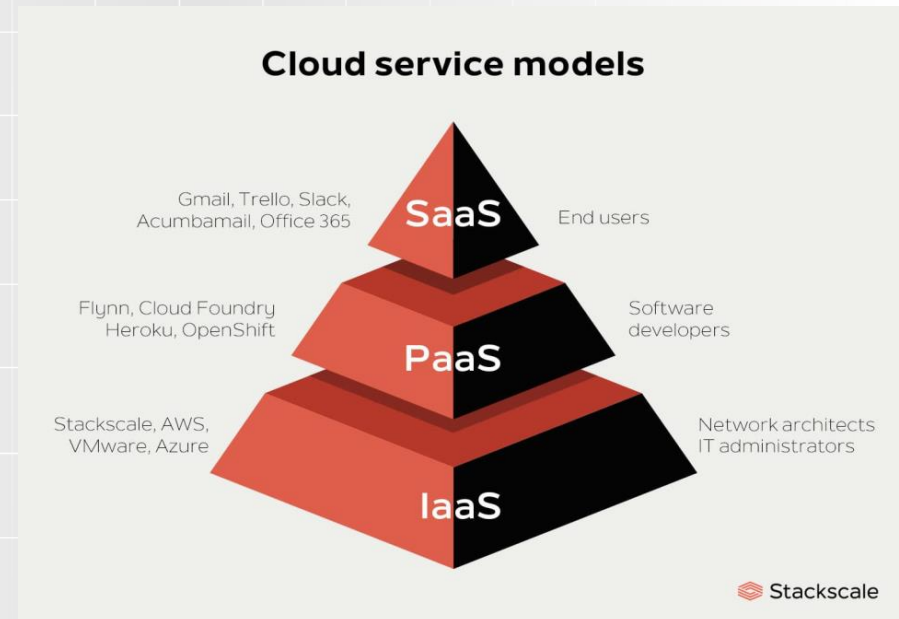
Infrastruktura jako usługa (IaaS)

- Model, w którym świadczone są maszyny wirtualne i serwery dla klientów do hostowania szerokiej gamy aplikacji i usług IT.
- Moc obliczeniowa, sieć i pamięć masowa dostarczane przez Internet
- Przykłady: Amazon EC2, Rackspace, Google Compute Engine



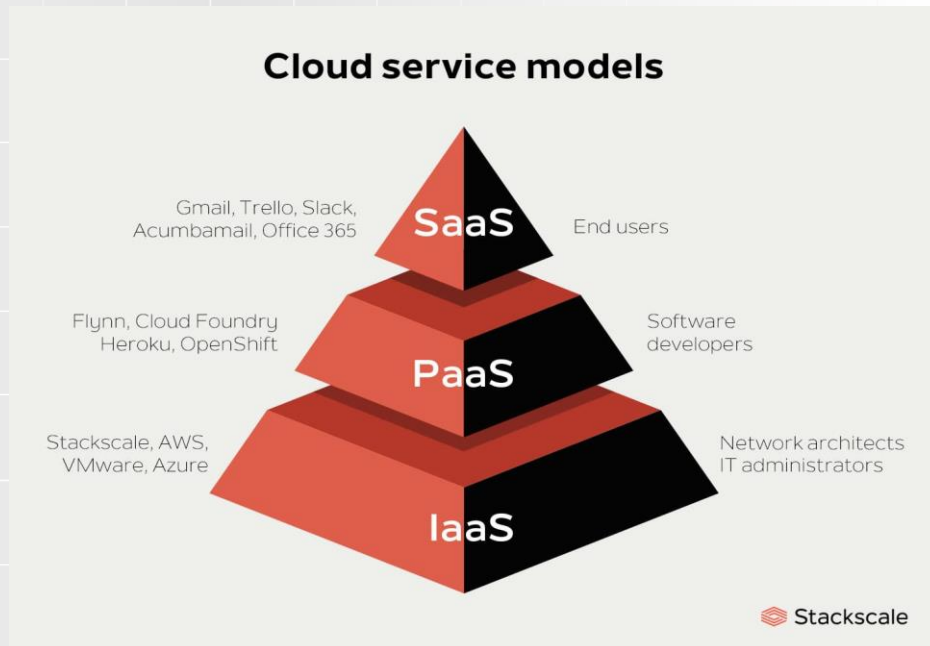
Platforma jako usługa (PaaS)

- Model, który zapewnia klientom wirtualną platformę do tworzenia niestandardowego oprogramowania.
- Narzędzia udostępniane przez Internet do tworzenia programów i aplikacji
- Przykłady: AWS Elastic Beanstalk, Microsoft Azure, Google App Engine



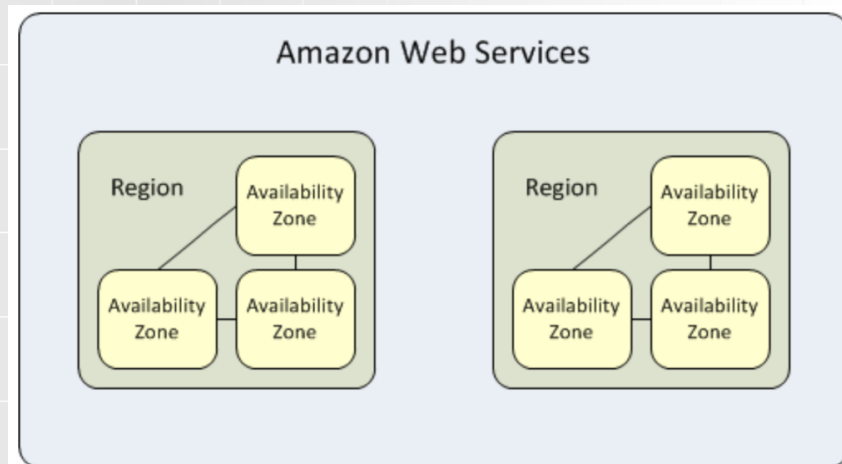
Oprogramowanie jako usługa (SaaS)

- Model udostępniający aplikacje korzystające z Internetu, które są zarządzane przez stronę trzecią.
- Aplikacje i programy dostępne i udostępniane przez Internet
- Przykład: Dropbox, Slack, Spotify, YouTube, Microsoft Office 365, Gmail



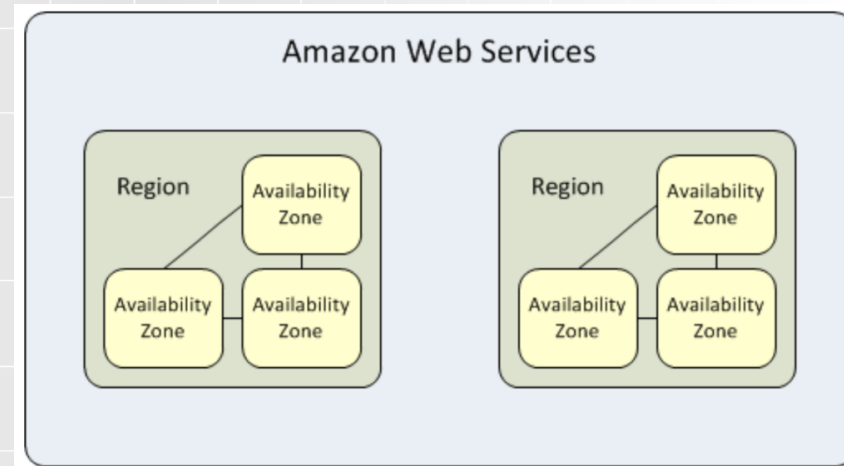
Region

- Obszar, w którym przechowywane są dane.
- Każdy region to osobny obszar geograficzny.
- Każdy Region został zaprojektowany tak, aby był odizolowany od innych Regionów.
- Podczas przeglądania zasobów (np. instancje EC2), widoczne są tylko te, które są powiązane z wybranym regionem.



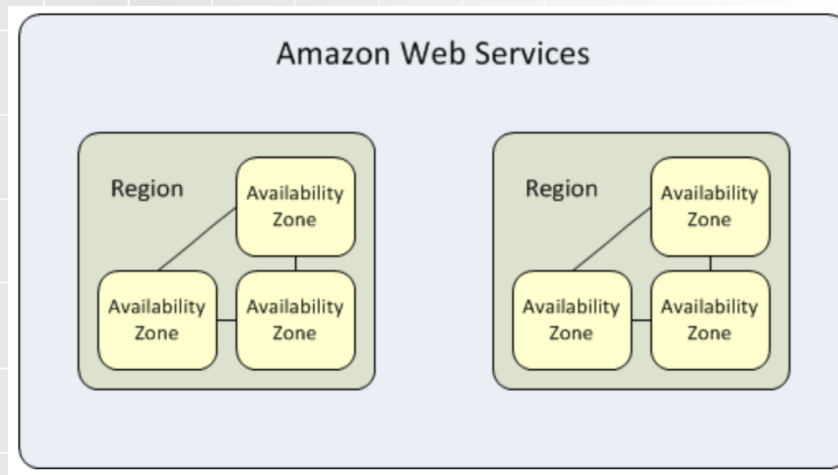
Strefa dostępności (Availability Zone)

- Jedno lub więcej centrów danych, w których znajduje się wiele serwerów.
- Każdy region ma wiele odizolowanych lokalizacji zwanych Strefami Dostępności.
- Każda strefa dostępności jest izolowana, ale strefy dostępności w regionie są połączone za pomocą łącz o małym opóźnieniu.
- Strefa dostępności jest reprezentowana przez kod regionu, po którym następuje identyfikator literowy, na przykład us-east-1a.



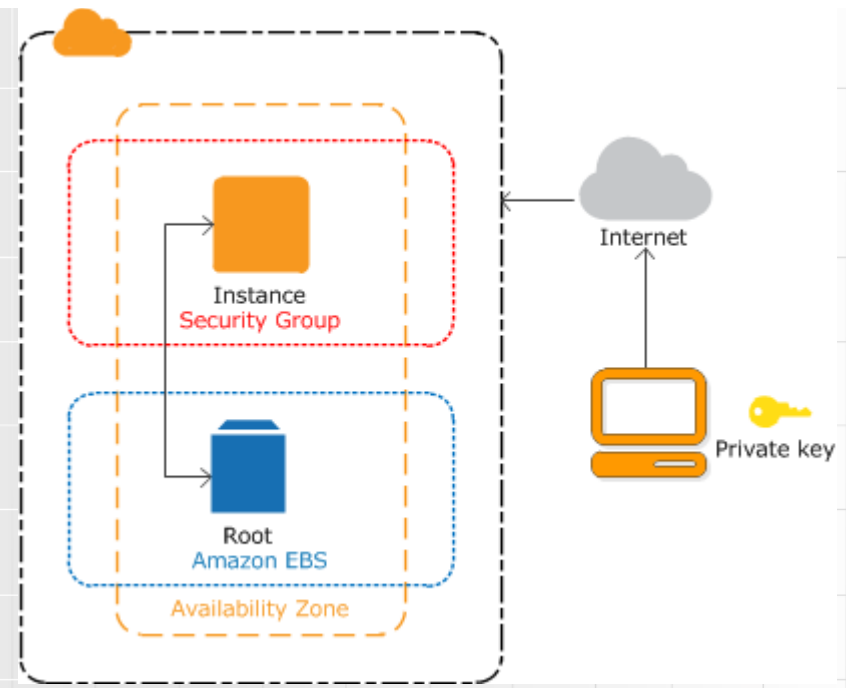
Serwery brzegowe (Edge locations)

- Centra danych AWS zaprojektowane w celu dostarczania usług z możliwie najmniejszymi opóźnieniami.
- Są bliżej użytkowników niż Regiony czy Strefy Dostępności, często w dużych miastach, więc odpowiedzi mogą być możliwe jak najszybsze.
- Tylko część usług dla których ma to znaczenie korzysta z lokalizacji brzegowych np. CloudFront, Route 53, Web Application Firewall, AWS Shield



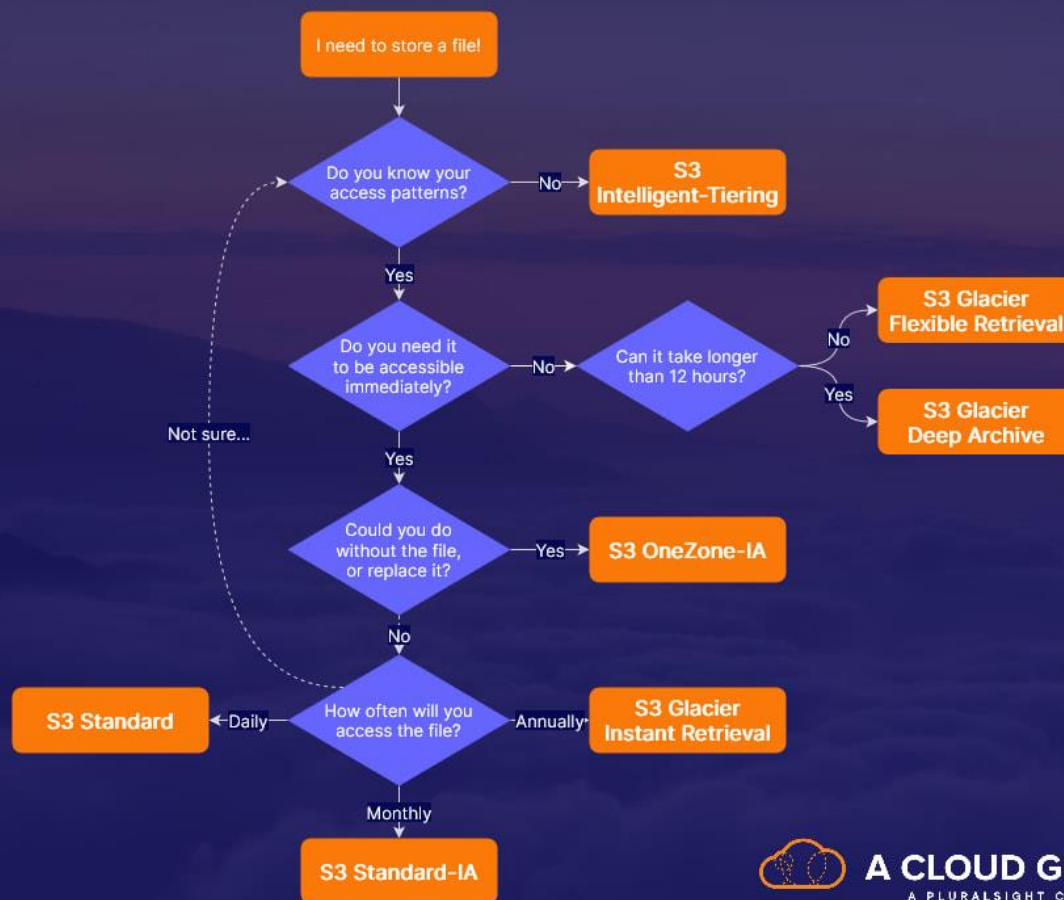
Amazon EC2 (Elastic Cloud Compute)

- Usługa która pozwala na stworzenie serwera w chmurze AWS
- AWS nazywa te serwery instancjami (instances)
- Użytkownik może tworzyć, uruchamiać i zamykać instancje serwera w razie potrzeby
- Zapewnia użytkownikom kontrolę nad położeniem geograficznym instancji



Which S3 Storage Class is right for me?

December 2021



A CLOUD GURU
A PLURALSIGHT COMPANY



Wrocław
University
of Science
and Technology

Dziękuję za uwagę