Konfiguracja usług w MS AZURE

Wymaganie wstępne: Proszę założyć konto **Azure for Students** (darmowe kredyty) lub konto Free Trial. **Uwaga:** Po zakończeniu każdego zadania należy usunąć grupę zasobów (Resource Group), aby uniknąć niepotrzebnego zużycia kredytów!

A: Podstawy (IaaS i Sieci)
Celem tego ćwiczenia jest zrozumienie podstawowych bloków budulcowych chmury: sieci wirtualnych i maszyn wirtualnych.

A1: Architektura sieciowa i maszyna wirtualna

Scenariusz: Firma potrzebuje bezpiecznego serwera dostępnego z Internetu, ale z ograniczonym dostępem.

1. **Resource Group:** Utwórz nową grupę zasobów o nazwie Lab1-RG. W polu region zaznacz (EUROPE) Poland Central
2. **VNET:** Skonfiguruj sieć wirtualną (VNet) z przestrzenią adresową 10.0.0.0/16.

A2: Utworzenie sieci wirtualnej (VNet)

1. W pasku wyszukiwania wpisz **Virtual networks** i wybierz usługę.
2. Kliknij **+ Create**.
3. **Zakładka Basics:**
    - **Resource group:** Wybierz istniejącą (np. Lab-WebSQL-RG) lub utwórz nową Lab-Network-RG.
    - **Name:** VNet-Lab-01.
    - **Region:** Ten sam, co grupy zasobów ((EUROPE) Poland Central).
4. **Zakładka IP Addresses (Kluczowa):**
    - Azure może domyślnie zaproponować adresację. Jeśli widzisz już wpisane 10.0.0.0/16, możesz go edytować lub usunąć (ikona kosza) i dodać własny, aby przećwiczyć.
    - **IPv4 address space:** Wpisz 10.0.0.0/16.
        - *Wyjaśnienie:* /16 jest maska sieci, co oznacza, że masz do dyspozycji 65,536 adresów IP (od 10.0.0.0 do 10.0.255.255).
5. **Konfiguracja Podsieci (Subnets):**
    - Nadal w zakładce IP Addresses, kliknij **+ Add subnet** na dole.
    - **Podsieć 1 (Publiczna/Frontend):**
        - **Subnet name:** Frontend-Subnet.
        - **Subnet address range:** 10.0.1.0/24.
        - *Wyjaśnienie:* /24 daje 256 adresów (realnie 251 dla użytkownika, bo Azure rezerwuje 5 adresów).
        - Kliknij **Add**.
    - **Podsieć 2 (Prywatna/Backend):**
        - Kliknij ponownie **+ Add subnet**.
        - **Subnet name:** Backend-Subnet.

        - **Subnet address range:** 10.0.2.0/24.
        - Kliknij **Add**.
6. Kliknij **Review + create** -> **Create**.

B: Zabezpieczenie sieci (Network Security Group)

Sama sieć to za mało. Musimy określić, jaki ruch jest dozwolony. Użyjemy do tego NSG (Network Security Group).

1. W wyszukiwarce wpisz **Network security groups**.
2. Kliknij **+ Create**.
3. **Basics:**

    - ✓ **Resource Group:** Ta sama co przy VNet.
    - ✓ **Name:** NSG-Frontend.
    - ✓ **Region:** Ten sam co VNet (Ważne!).
    - ✓ Kliknij **Review + create** -> **Create**.

4. Po utworzeniu, kliknij **Go to resource**.

C: Tworzenie reguł ruchu (Inbound Rules)

Domyślnie NSG blokuje cały ruch przychodzący z Internetu. Musimy "zrobić dziurę" dla ruchu HTTP.

1. W menu po lewej (sekcja Settings) wybierz **Inbound security rules**.
2. Kliknij **+ Add**.
3. Skonfiguruj regułę dla strony WWW:
    - **Source:** Any (każdy z Internetu).
    - **Source port ranges:** *.
    - **Destination:** Any.
    - **Service:** Wybierz HTTP (Azure automatycznie ustawi port 80).
    - **Action:** Allow.
    - **Priority:** 100 (im niższy numer, tym ważniejsza reguła).
    - **Name:** Allow-HTTP.
    - Kliknij **Add**.

Dodaj drugą regułę, która pozwala na ruch SSH (port 22), ale **Source** ustaw na My IP Address (Azure automatycznie wykryje Twoje IP). To dobra praktyka bezpieczeństwa.

D: Podpięcie NSG do podsieci (Association)

Przypnij **NSG** do konkretnej podsieci.

1. Będąc w widoku swojego NSG (NSG-Frontend), w menu po lewej wybierz **Subnets**.
2. Kliknij **+ Associate**.
3. **Virtual network:** Wybierz VNet-Lab-01.
4. **Subnet:** Wybierz Frontend-Subnet.

5. Kliknij **OK**.

W efekcie każdy serwer (Maszyna Wirtualna), który utworzysz w podsieci Frontend-Subnet, będzie automatycznie chroniony przez te reguły (otwarte HTTP, otwarte SSH tylko dla Ciebie). Podsieć Backend-Subnet pozostaje bezpieczna (brak dostępu z zewnątrz), dopóki nie przypniesz do niej osobnego NSG.

1. **Maszyna Wirtualna:** Utwórz maszynę wirtualną (VM) z systemem Ubuntu lub Windows Server w podsieci Frontend. Wybierz rozmiar B1s lub B1ls (najtańsze).
2. **NSG (Network Security Group):** Skonfiguruj reguły bezpieczeństwa:
3. Zezwól na ruch SSH (port 22) lub RDP (3389) **tylko** ze swojego adresu IP (My IP).
4. Zezwól na ruch HTTP (port 80) z całego Internetu.
5. **Weryfikacja:** Połącz się z maszyną i zainstaluj prosty serwer WWW (np. Apache/Nginx lub IIS), aby sprawdzić, czy strona wyświetla się w przeglądarce.

E: Usługi platformowe (PaaS - Web & Database)

Przejście od zarządzania systemem operacyjnym do zarządzania samą aplikacją i danymi.

F: Wdrożenie dwuwarstwowej aplikacji Web

Scenariusz: Należy uruchomić sklep internetowy bez martwienia się o aktualizacje systemu operacyjnego serwera.

1. **Azure SQL Database:** Utwórz bazę danych SQL (w modelu DTU Basic lub Serverless, aby zminimalizować koszty). Skonfiguruj zaporę (Firewall) serwera bazy, aby zezwolić na dostęp usługom Azure.
2. **App Service Plan:** Utwórz plan w wersji darmowej (F1) lub podstawowej (B1).
3. **Web App:** Utwórz aplikację Web App.
4. **Deployment:** Wdróż prostą aplikację (może to być przykładowy kod z GitHub dostarczony przez Microsoft, np. "Wingtip Toys" lub prosta aplikacja .NET/Python wyświetlająca "Hello World").
5. **Zmienne środowiskowe:** Skonfiguruj *Connection String* do bazy danych w sekcji "Configuration" aplikacji Web App, zamiast trzymać hasło w kodzie (Best Practice).

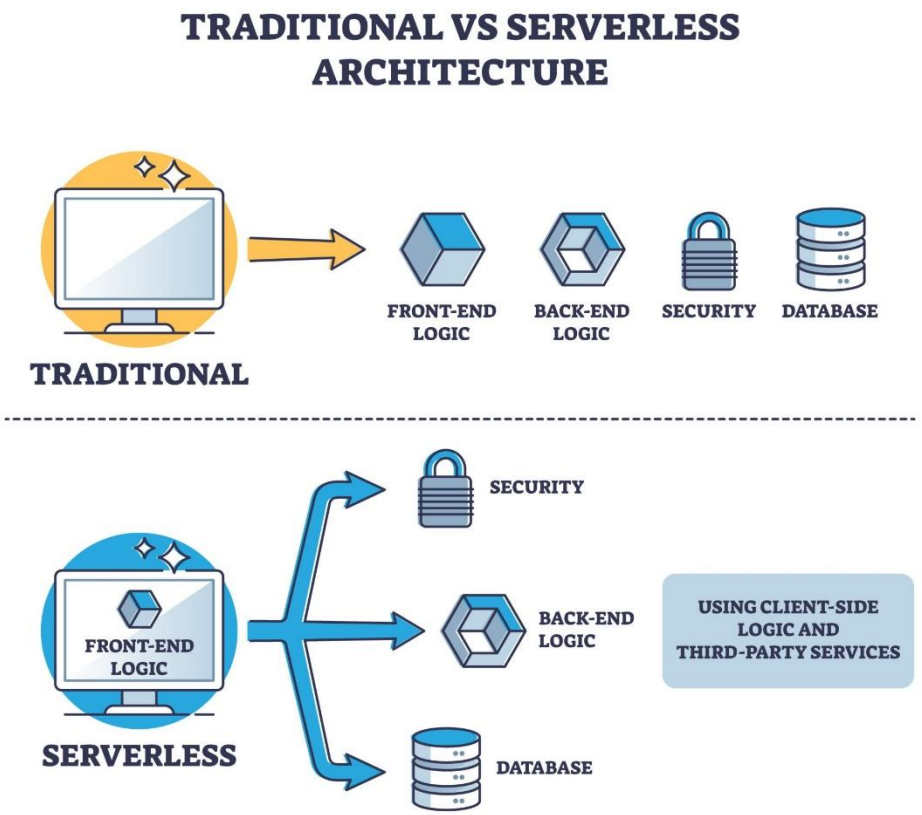
G: Storage i Serverless

Wprowadzenie do nowoczesnych architektur sterowanych zdarzeniami (Event-driven) i taniego hostingu.

3: Statyczna strona i przetwarzanie obrazów

Scenariusz: Hosting strony marketingowej oraz automatyczne tworzenie miniatur (thumbnails) przesłanych zdjęć.

1. **Storage Account:** Utwórz konto magazynu. Włącz funkcję "Static Website" i wgraj tam prosty plik index.html.
2. **Kontenery:** W tym samym koncie utwórz kontenery images-in oraz images-out.
3. **Azure Function:** Utwórz funkcję (Consumption Plan - płatne tylko za użycie), która jest wyzwalana (Trigger) pojawieniem się nowego pliku w kontenerze images-in.
4. **Logika:** Funkcja powinna pobrać obraz, zmienić jego rozmiar (np. używając biblioteki Pillow dla Python lub ImageSharp dla C#) i zapisać wynik w images-out.
5. **Test:** Wgraj zdjęcie do pierwszego folderu i sprawdź, czy miniatura pojawiła się w drugim.



Źródło: Getty Images

H: Bezpieczeństwo i tożsamość (Governance)

Kluczowe umiejętności dla administratorów – zarządzanie dostępem i politykami.

I: Key Vault i zarządzanie tożsamością (Entra ID)

Scenariusz: Deweloperzy nie mogą widzieć haseł do bazy danych, a aplikacja musi pobierać je bezpiecznie.

1. **Azure Key Vault:** Utwórz sejf (Key Vault). Dodaj do niego "Secret" (np. hasło do bazy danych z Zadania 2).
2. **Managed Identity:** Włącz tożsamość zarządzaną (System Assigned Managed Identity) dla aplikacji Web App lub Funkcji z poprzednich zadań.
3. **Access Policies:** W Key Vault skonfiguruj politykę dostępu (Access Policy) lub RBAC, aby **tylko** ta konkretna aplikacja (jej tożsamość) miała prawo do odczytu sekretów (GET/LIST secrets).
4. **Weryfikacja:** Zmodyfikuj kod aplikacji, aby pobierał hasło z Key Vault zamiast ze zmiennych środowiskowych.

J: Konteneryzacja (Docker w Chmurze)

Wstęp do mikroserwisów bez pełnego klastra Kubernetes (zbyt drogiego/złożonego na start).

J1Azure Container Registry i Container Instances

Scenariusz: Firma chce przenieść aplikację legacy zapakowaną w Dockerze do chmury.

1. **ACR (Azure Container Registry):** Utwórz prywatny rejestr kontenerów.
2. **Docker Build & Push:** Na lokalnym komputerze (lub w Azure Cloud Shell) zbuduj prosty obraz Dockera (np. z plikiem Dockerfile zawierającym prosty serwer Node.js) i wypchnij (push) go do swojego ACR.
3. **ACI (Azure Container Instances):** Uruchom ten kontener bezpośrednio w chmurze, korzystając z obrazu znajdującego się w Twoim prywatnym rejestrze ACR.
4. **Weryfikacja:** Wejdź na publiczny adres IP kontenera i sprawdź działanie aplikacji.