Opis sytuacji:

Jesteś architektem w dużej firmie (20 000 pracowników, ponad 5000 maszyn wirtualnych, ponad 200 maszyn fizycznych pod środowiskiem wirtualnym), która podjęła decyzję o migracji do Chmury. W firmie jest dość znaczna ilość systemów „legacy” i pierwszy kroki migracji to będzie migracja typu Lift&Shift, gdzie głównie wykorzystasz środowiska maszyn wirtualnych, ale postarasz się zautomatyzować cały proces powoływania i zarządzania środowiskami.

Zadania:

* #TYDZIEN4.1 Na bazie swoich doświadczeń i podstawowej wiedzy o maszynach wirtualnych oraz Scale Set zaproponuj kilka kroków, które byś zrealizował w ramach takiej migracji. Przy każdym kroku napisz maksymalnie 2-3 zdania, jakie czynności w takim kroku byś zrealizował.

Pamiętaj o ograniczeniach w chmurze, limitach oraz ograniczeniach co do tego, jakie parametry wirtualnych maszyn, możesz uzyskać w Azure.

Wszelkie dodatkowe komentarz mile widziane! Wiem, że masz mało danych, ale… wiesz, ja też mam zawsze za mało danych do migracji, co nie powinno mnie powstrzymać, by zaplanować proces.

* #TYDZIEN4.2 Virtual Machine Scale Set

VMSS nie są często używane w projektach. W ramach zadania nr. 2 napisz mi proszę do jakich warstw aplikacji użyłbyś Scale Set a następnie spróbuj za pomocą Azure CLI zbudować swój prosty Scale Set.

#TYDZIEN4.1

1. Mierzę siły na zamiary, także pierwszy krok to zbieranie informacji (a najlepiej sformowanie specjalengo zespołu) na temat przeprowadzania takich migracji.
2. Zakładana jest migracja typu Lift&Shift, ale z możliwością wykorzystania VMSS tam gdzie przyniesie to uzasadnione korzyści (i lepiej pozwoli wykorzystać możliwości chmury). Patrząc na rozmiar istniejącej infrastruktury uważam, że bardziej bezpieczne będzie migrowanie kolejnych aplikacji w osobnych krokach, zamiast tzw. „big bang”.
3. Zaczynamy od wewnętrznej inwentaryzacji na poziomie ogólnym:
   1. Jakie aplikacje należy zmigrować
   2. Jak wygląda obecna infrastruktura
   3. Zależności pomiędzy powyższymi z uwzględnieniem systemów operacyjnych na których działają aplikacje
4. Punkt 3 powinien pozwolić na stworzenie docelowej architektury (również na poziomie ogólnym) i zweryfikowanie czy na tym etapie nie pojawiły się już pierwsze znaczące wyzwania.
5. Sprawdzamy najbardziej krytyczne elementy aplikacji, aby w odpowiedni sposób dobrać dla nich usługi (opierałbym się również na pryncypiach projektowania aplikacji w Azure https://docs.microsoft.com/en-us/azure/architecture/guide/design-principles/)
6. Następny krok to na bazie punktów 3, 4 i 5 wybranie kolejności w jakiej odbywać się będzie migracja, zaczynając od aplikacji które będą pozwalały się przenieść praktycznie 1-1. Tutaj również warto zastanowić się które z aplikacji mogą zyskać najwięcej na możliwościach oferowanych przez chmurę, może pozwoli to na rozwiązanie do tej pory często występujących problemów (automatyczne skalowanie, wysoka dostępność, zmniejszenie latency poprzez wykorzystanie wielu regionów)
7. Punkty 3-6 powtarzam kilka razy, schodząc na niższe poziomy szczegółowości i zbierając kolejne wnioski. Niższe poziomy szczegółowości powinny zawierać takie elementy jak:
   1. Wersje OS, patche, zainstalowane runtime, dodatkowe biblioteki
   2. Szczegóły insfrastruktury sieciowej i wymaganej komunikacji
   3. Dane dotyczące zużycia zasobów przez aplikacje
   4. Polityki backupowania, HA & DR, SLA
   5. Storage
   6. Automatyzacja procesów, schedulery, agenci etc.
   7. Wszelkie aspekty prawne dotyczące np. polityki w danym państwie, branży, składowanych danych

Wszystkie powyższe pozwolą na dobranie właściwych usług i ich konfiguracji.

Dodatowo, mimo tego że robimy Lift&Shift warto skorzystać z takiej analizy wielowymiarowo i wskazać na przyszłość które elementy warto zmigrować potem do PaaS.

Do tego dodam, że znalazłem taki ciekawy pdf, który jest swego rodzaju przewodnikiem przygotowującym do tego typu wyzwań: <https://azure.microsoft.com/en-us/resources/azure-virtual-datacenter-lift-and-shift-guide/>

#TYDZIEN4.2

Warstwy aplikacji gdzie widzę zastosowanie VMSS:  
- aplikacje / warstwy aplikacji stateless (np. jakiś frontend do apki webowej)

- węzły obliczeniowe (np. przy skalowaniu procesów ETL / ELT)