

Produkcyjna Landing Zona w dwa tygodnie?



@RogalaPiotr



justcloud



Nordcloud

an IBM Company

FIELD MPH
20

Piotr Rogala

Working in



Principal Architect, Azure

Blog



MVP Azure



Group leader



Microsoft Azure
User Group Poland



@RogalaPiotr



linkedin.com/in/rogalapiotr

Agenda

1. Wprowadzenie
 - a. Rozwiązania
 - b. Architektura
 - c. Napotkane błędy
2. Wnioski
3. Podsumowanie



[WRO] 25 spotkanie Microsoft Azure User Group Poland we Wrocławiu



Hosted By
Piotr R. and Michał J.

m meetup



Details

Serdecznie zapraszamy na 25 spotkanie Microsoft Azure User Group Poland we Wrocławiu, które odbędzie się 30 Października we Środę!

Uwaga! nowa lokalizacja!

📍 Miejsce spotkania: **GlobalLogic**, ul. Strzegomska 48a, 53-611 Wrocław
🔗 <https://maps.app.goo.gl/4DpcYRd4B7yv4qPd8>

W trybie ciągłym zachęcamy do zgłoszania swoich wystąpień na kolejnych meetupach.
CFP jest dostępne tutaj ↗ <https://sessionize.com/AzureWroclawMeetups>

Organizer tools ▾



Microsoft Azure User Group Poland

Public group ?

⌚ Wednesday, October 30, 2024
6:00 PM to 9:00 PM CET
[Add to calendar](#)

📍 GlobalLogic
Strzegomska 48a · Wrocław
How to find us
Meetup jest w budynku GlobalLogic,
ul. Strzegomska 48a, 53-611 Wrocław



Google

Report this event



Project

intro

Projekt w spadku

- Duży Niemiecki start-up
- Dostawca duża Niemiecka firma
- Rok pracy dostawcy na Landing Zone
- Jedno spotkanie na handover
- Dokumentacja - jedna strona na Confluence (czyli brak)



Deadline

- 2 tygodnie na uruchomienie środowisk
- Go live za miesiąc stąd jest potrzebne środowiska dla Dev Teams
- W projekcie 3 osoby na około 80% czasu



Narzucone rozwiązania

- Azure (Landing Zone)
- Terraform (Landing Zone)
- Terragrunt (Landing Zone)
- GitLab **export** pipeline
- GitHub **import** pipeline (Landing Zone)
- HashiCorp Vault (Apps)
- Kubernetes Service + Argo CD (Apps)
- Postgres (Apps)

Obecny / dostarczony kod

- Dwa repozytoria
 - Terragrunt
 - Terraform
- Podział na środowiska, hub, ops, audit, env1, env2...
- Bardzo dużo kodu / modułów
- Brak connectivity
- Brak routes / peerings
- Brak Application Gateways
- Słabe security - statefiles, aks, key vault's (public endpoints)



Założenia / problemy / wdrożenie

- Użycie obecnego kodu
 - Dużo braków konfiguracyjnych
- Wdrożenie środowisk od zera
 - Brak know-how
 - Przygotowanie planu i wdrażanie kolejno każdego środowiska
- Łączenie HashiCorp vault z Terraform i Argo CD
- Wdrażanie baz danych na Postgres z Terraform
- Nie działająca sieć wewnętrzna
- Deweloperzy sami zarządzają aplikacjami poprzez Argo CD
- Wiele wdrożeń w tym samym czasie na Agentach GitHub

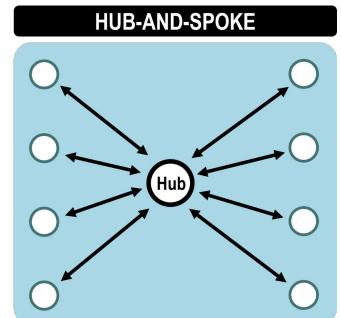
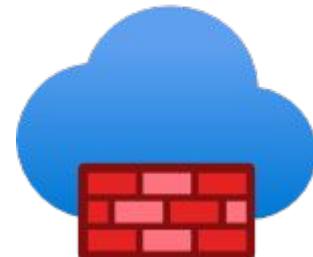
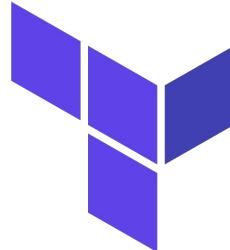
Narzucone rozwiązania (przegląd)

- Robić wszystko od zera?
 - Po co?
 - Czy to co mamy nadaje się do rozbudowy?
 - Terraform
- Brak czasu na zmiany używanych rozwiązań
 - Sens wprowadzania zmian?
- Skupienie się na "delivery"



Architektura / Decyzje

- Rozwijamy obecny kod
- Topologia Hub and Spoke
- Dostępność z Internetu przez AFW do LZ i Apps
- Wszystkie usługi odizolowane od publicznego dostępu
- Usługi wychodzą na świat przez Azure Firewall
- IaC - Terraform / Terragrunt
- GitHub z dostępym kodem od poprzedniego dostawcy



Terragrunt

Struktura folderów i plików była stworzona pod GitLab z wykorzystaniem Terragrunt i Terraform.



Formatting hcl files

You can rewrite the hcl files to a canonical format using the `hclfmt` command built into `terragrunt`. Similar to `terraform fmt`, this command applies a subset of [the OpenTofu/Terraform language style conventions](#), along with other minor adjustments for readability.

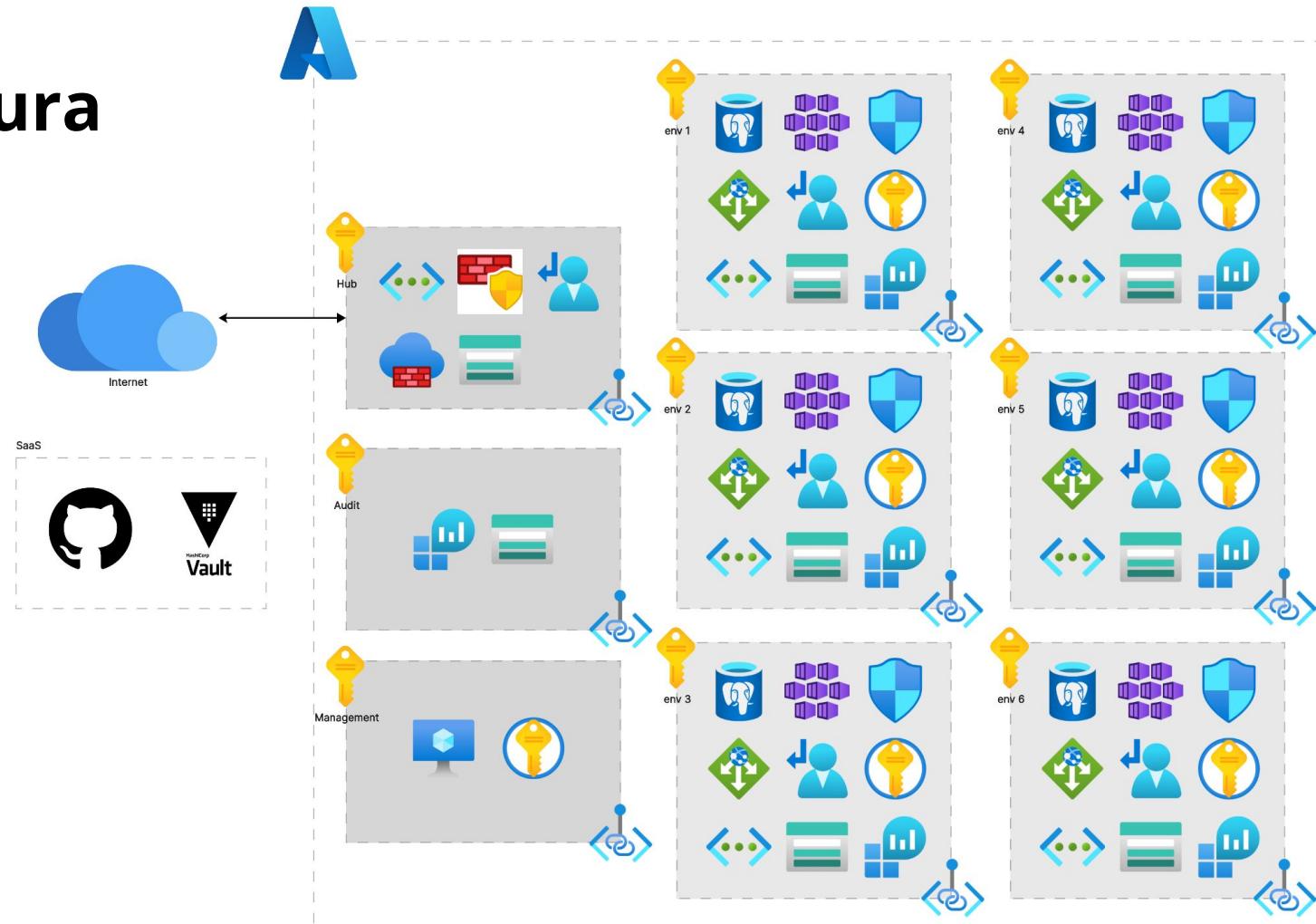
This command will recursively search for hcl files and format all of them under a given directory tree. Consider the following file structure:

```
root
├── terragrunt.hcl
├── prod
│   └── terragrunt.hcl
└── dev
    └── terragrunt.hcl
└── qa
    ├── terragrunt.hcl
    └── services
        ├── services.hcl
        └── service01
            └── terragrunt.hcl
```

If you run `terragrunt hclfmt` at the `root`, this will update:

- `root/terragrunt.hcl`
- `root/prod/terragrunt.hcl`
- `root/dev/terragrunt.hcl`
- `root/qa/terragrunt.hcl`
- `root/qa/services/services.hcl`
- `root/qa/services/service01/terragrunt.hcl`

Architektura



Azure Capacity issues - West Europe

- VM size - D-series capacity issues
- Azure Networking – Global WAN issues

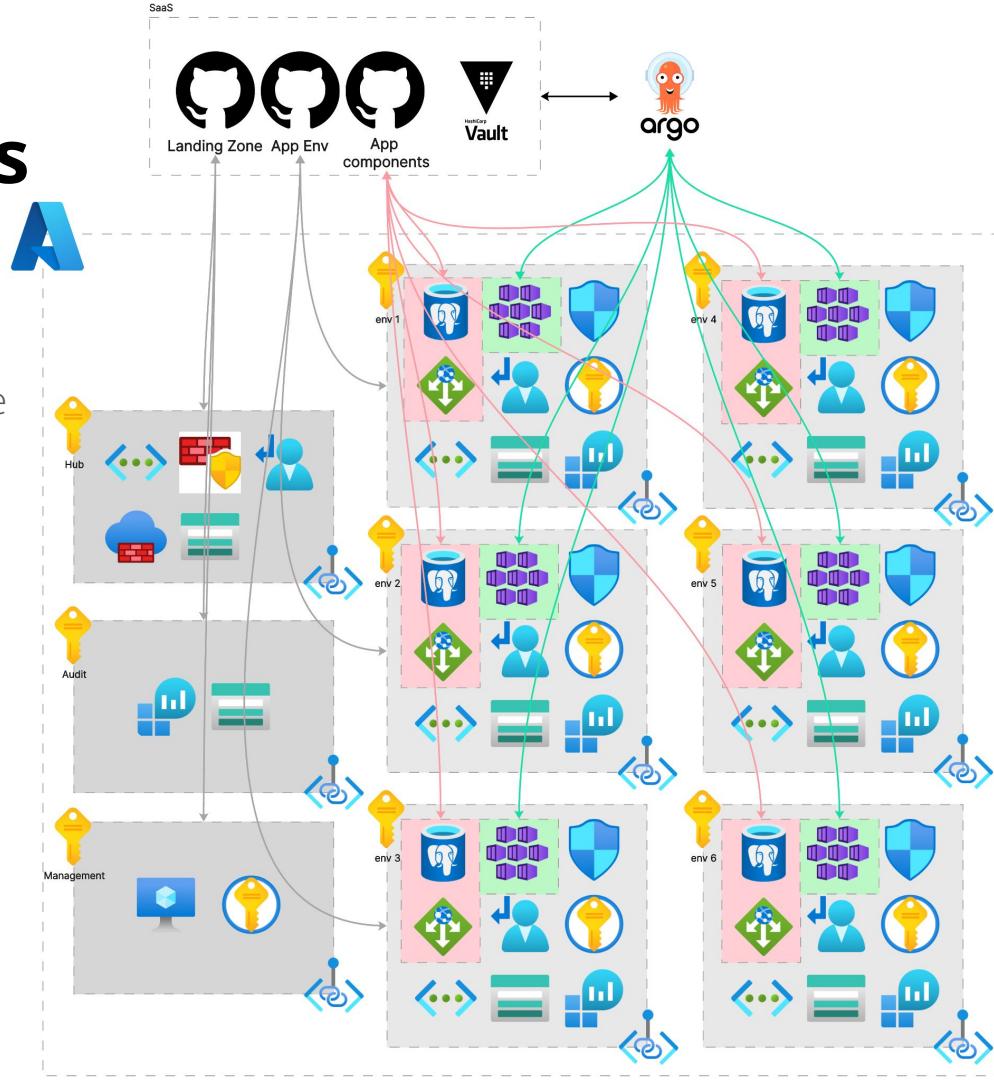
Przyspieszenie prac nad drugim regionem (HA)



<https://azure.status.microsoft/en-us/status/history/>

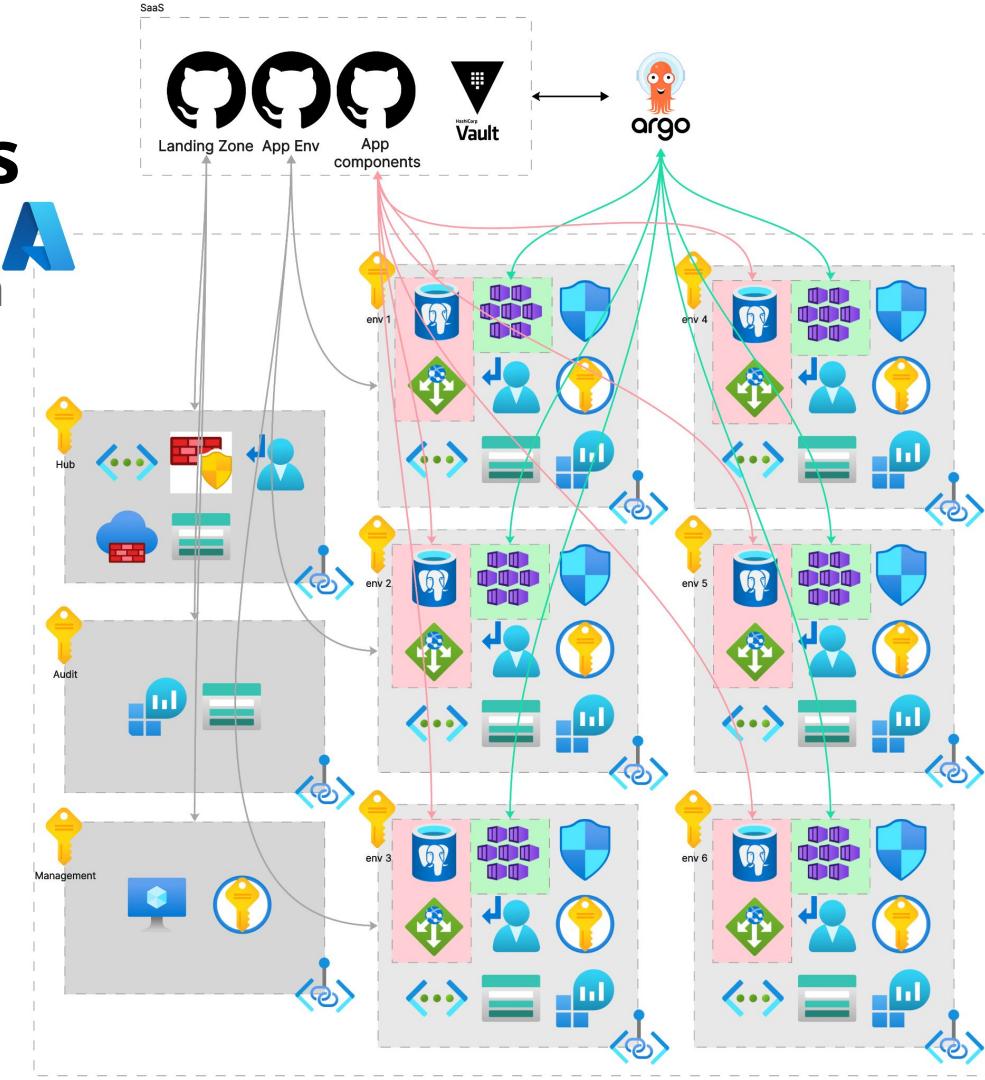
GitHub pipelines/repos

- Landing Zones
 - Podział per Subskrypcja
 - Podział na Landing Zone oraz Env type
- Approval gates
- Podział na environments w GH
- Podział SPN's na prod / nonprod
- Code security
 - Skanowanie kodu Dependabot
 - tfscan



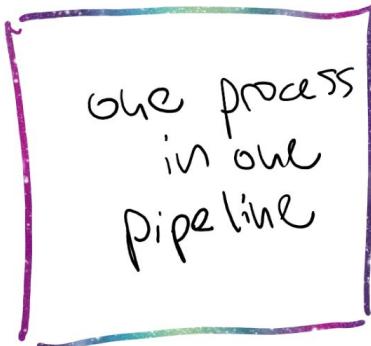
GitHub pipelines/repos

- Małe moduły zasobów w terraform
- Wdrożenia per Landing Zone
- Wdrożenia per środowisko app
 - Wiele configów terragrunt
 - Jeden main.tf z opisami środowisk app
- Zależności wdrożeń
 - App Gateway
 - ArgoCD
 - Postgres
- Podział odpowiedzialności
 - App Team
 - Ops Team
 - Security Team

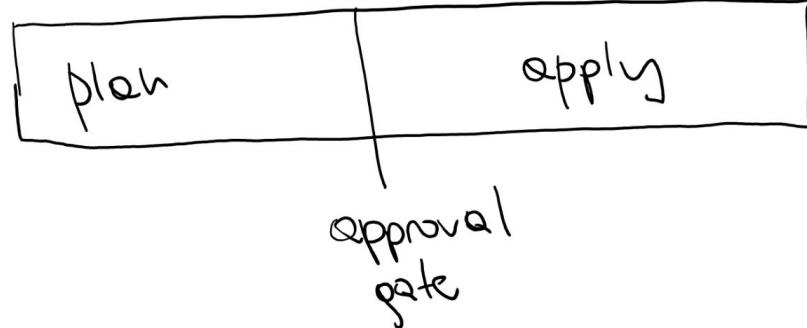


Pipelines

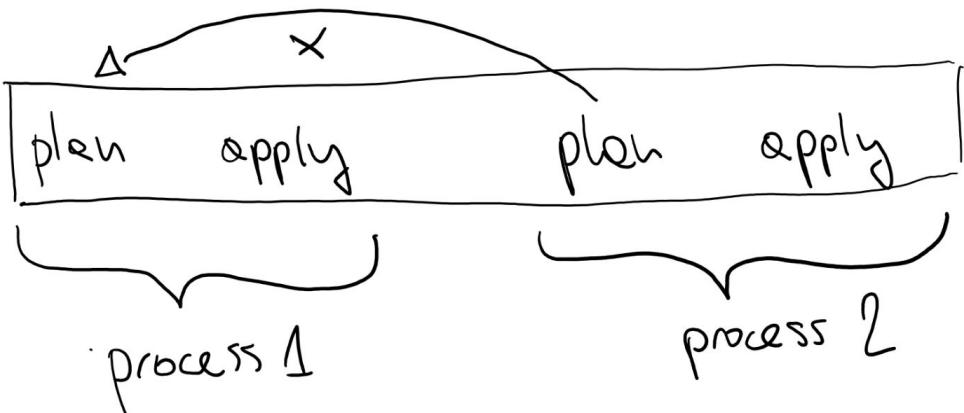
Problem z
podzielonym
wdrożeniem w
Terraform



normal task ci/cd



Complex task ci/cd



Napotkane błędy przy realizacji

- Problemy w Azure West Europe 🔥
 - Wymagane wiele środowisk - brak capacity w Azure
 - Brak zasobów jak: Synapse, Kubernetes Service
 - Problem z Azure Firewall
 - Brak HA (multi-region)
- Route all to Azure Firewall 🔥
 - Logi ruchu sieciowego są nieczytelne
- Wdrożenie kolejnego regionu 🔥
 - North Europe? Nie jest rozwiązaniem
 - Przebudowa Terraformu
 - Większe zmiany w modułach sieci wewnętrznej i naming convention
- Instalacja Argo CD per cluster 🔥
 - Co jeśli np. argocd-(dev|test|prod).customer.com i nowy cluster w nowym regionie?

Wnioski

- Wspólne wdrożenie dla LZ aplikacyjnej +
- Problemy ze state file -
 - Wiele podziałów konfiguracyjnych powoduje chaos
- Podział środowisk oraz Argo CD na wiele osobnych instancji -
- Niska elastyczność rozbudowy o nowy region -
- Złe nastawienie do automatyzacji, wszystko na siłę tworzone w Terraform -
 - Postgres
 - HashiCorp Vault
 - ArgoCD
 - Helm's
- Problemy z nazewnictwem -
- Narzędzia: GitHub, ArgoCD, Terraform +
- Podział odpowiedzialności, wiele środowisk, niezależne wdrożenia +



Podsumowanie

- Czy 2 tygodnie to wystarczająco? 📆
- Wymagane większe zaangażowanie zespołu 🧑
- Skupienie się na zmianach przyrostowych 📈
- Pozostawienie zmian, które nic nie wnoszą 🛡️
- Maksymalizacja usług PaaS 💼
- Optymalne podziały na wdrożenia LZ (Hub, Apps, Dev Apps) 🏠
- Landing Zone multi region (naming convention*) 🌎
- Miksować rozwiązania dbając o dobry opis i dokumentację 📄



Questions?



JustCloud.pl

O mnie

Blog

Meetup's

Kontakt



Kontakt

Jeśli chcesz się ze mną skontaktować to możesz to zrobić za pomocą tego formularza lub poprzez [LinkedIn](#), do zobaczenia!

E-mail

Message

Send message



Azure Workshop Wrocław - Learn & Fun



Organizatorzy

Jesteś zainteresowany tematyką Microsoft Azure?



Jeśli tak to zapisz się do listy mailingowej, aby dostać powiadomienie o nadchodzących wydarzeniach.



Piotr Rogala

Nordcloud, an IBM Company

Senior Cloud Architect Lead | Microsoft MVP



Michał Jankowski

Objectivity

Quantum Computing Enthusiast | Lead Technologist / Solution Architect at Objectivity | Microsoft MVP

<https://www.subscribepage.com/wroclaw-newsletter-workshops>