



SKÝVAFNIR



# Formlega skilgreindir innviðir (IaC)

(eða af hverju ég myndi alltaf velja Terraform)

Skýrslutæknifélag Íslands  
Rekstur tölvukerfa  
Hádegisverðarfundur - 8. Maí 2024

Gissur Þórhallsson

# Yfirlit

1. Hvernig enduðum við hér?
2. Formlega Skilgreindir Innviðir
3. Af hverju Terraform?
4. Lykilatriði góðrar IaC vegferðar

# 2006 – Loftmyndir

- Gissur skrifar kortasjá í Javascript

# 2006 – Loftmyndir

- Gissur ~~skrifað~~ græjar kortasjá í Javascript
- Virkaði FÍNT á minni vél
- Vantar keyrsluumhverfi
- Sendi email og panta þjón í rekka hjá EJS
- Þæli ekki meir í því, í bili

**4 – 6 vikum síðar**

# 2006 – Loftmyndir

- Dagur 0
- SSH-a mig inn á vélina
- Set upp Apache httpd
- Kópera kóða inn á vél (*ekkert source control*)
- Allt virkar!
- Svaka glaður

# 2006 – Loftmyndir

- Dagur 1
- Ekkert virkar
- Kortasjáin birtist
- Engin gögn
- Hver þremillinn!

# 2006 – Loftmyndir

- Dagur 1
- Set upp allskonar
  - Postgres / PostGIS / Tomcat / GeoServer
- ... í höndunum: `./configure && make && make install`  
*(googla villuskilaboð, gera græjur, keyra aftur, ad nauseam)*



**Eftir 1 – 5 daga af því**



# 2007 – Loftmyndir

- Þarf að uppfæra [eitthvað]
- Man ekki hvernig ég gerði neitt
- Skjalaði ekki neitt
- Ligg yfir bash-history á EJS þjóninum
- Redda þessu á endanum, úrvinda og dapur.

**Nóg um það**

# 2012 – Square Enix

- Uppgötva Source Control
- Django keyrandi í **AWS** á móti Postgres
  - Cloudfront / ELB
  - 2 VMs (EC2)
  - Postgres (RDS)
  - Network (VPCs)

# 2012 – Square Enix

- AWS innviðir settir upp með **python/boto3** kóða
- 3 umhverfi (DEV/TEST/PROD)
- Ítrað hratt á móti DEV
  - Hægar á móti TEST
  - Uppfærslur sjaldan vandamál
- Boto3 kóðabasinn var „ónæmur fyrir breytingum“
- En virkaði bara

# 2014 - Næsta fjör

# 2014 – Fibetco

Stofna startup með samverkamönnum frá Square Enix

- Heimasíða og vefþjónustur.
  - *Angular, NodeJS, Python, Postgres. Alltaf Postgres.*
- Allt í AWS, mikill hraði, allt clickopsað (reddum því seinna™)
- Byrjum að vaxa smá en það er allt sífelld að fara á hliðina.
- Erfiðleikar með útgáfu og stöðugleika.
- Síminn rauðglóandi.
- Peningarnir klárast.
- Sjaldan verið svona létt



# 2017 - Fluttur heim

# 2017 – WUXI NextCODE

- Terraform
- Kubernetes
- 30+ umhverfi (mix af DEV/TEST/PROD)
- Mikið cloud
- Mikið automation
- Mikið gaman

# 2018 – WUXI NextCODE

- Skrifuðum þjónustu og þurftum að deploya henni.
- „DevOps“ teymi til staðar sem gaf mér sandkassa.
  - „þarf þjónustan að tala við internetið?“
  - „þarf þjónustan að tala við grunn / NFS-ið?“
- Gissur pattern-matchaði eins og vindurinn, og lærði „óvart“ Terraform.
- **Fatta að síminn er hættur að hringja á kvöldin**
  - Búinn að lenda í og leysa öll vandamálin á DEV / TEST

**Og einni Terraform / Kubernetes vinnu  
síðar (*Mainframe Industries*) ...**

# 2024 – Skývafnir

- Búinn að drekka allt Terraform cool-aidið
- Kominn hingað upp í púlt
- Er vonandi ekki búinn með tímann.

# Formlega Skilgreindir Innviðir

# IaC Landslagið

# IaC Landslagið

- Uppsetning og rekstur innviða  
*(e. infrastructure provisioning and management)*
- Útgáfustjórnun  
*(e. release management)*
- Stillingastjórnun  
*(e. configuration management)*
- Vinnslustýring  
*(e. workload orchestration)*



# IaC Landslagið – smá sápuópera

1. Hashicorp breytir Terraform leyfinu
  - Mozilla Public License -> BSL
2. Ákveðinn hluti TF samfélagsins fer á hliðina
3. OpenTofu verður til.
4. [alls konar fjör á milli þessara aðila]
5. IBM kaupir Hashicorp (og þ.mt. Terraform)
6. Og hér erum við

# Af hverju Terraform?

# Af hverju ekki eitthvað annað?

## CloudFormation

- Eingöngu AWS
- Lítið (*og minnkandi?*) samfélag
- Þykir almennt ekki frábært

## Bicep

- Eingöngu Azure (ARM)
- Lítið, en vaxandi samfélag
- Nýlegt

# Af hverju ekki eitthvað annað?

- **Pulumi / Vendor SDKs / CDK / etc.**

- Meira Developer focused
- Flóknara

- **Terragrunt**

- Terraform orchestrator
- Þunn skel yfir Terraform.
- Fólk þarf að læra á bæði, sem flækir onboarding

# Af hverju þá Terraform?

- Einfalt
- Læsilegt  
(af „*Homo Informaticus*“)
- Sjálfvirk skjölun
- ...sjá mynd

```
# main.tf
```

```
locals {  
  tags = {  
    instance = "demo"  
    org_code = "sky"  
    tier      = "dev"  
    owner    = "gissur@skyvafnir.cloud"  
  }  
}
```

```
resource "azurerm_virtual_network" "this" {  
  name                = module.defaults.resource_name  
  resource_group_name = var.resource_group_info.name  
  location            = var.resource_group_info.location  
  address_space       = [var.virtual_network_cidr] # "10.0.0.0/16"  
  
  tags = module.defaults.tags  
}
```

```
$ terraform plan
```

```
# module.sky-demo.module.network_setup.azure_rm_virtual_network.this will be created
+ resource "azurerm_virtual_network" "this" {
  + address_space      = [
    + "10.10.0.0/16",
  ]
  + location            = "northeurope"
  + name                = "sky-demo-vnet-dev"
  + resource_group_name = "sky-demo-rg-dev"
  + tags                = {
    + "_iac-part-of-module" = "network-base"
    + "_iac-provisioned-by" = "skyvafnir-terraform"
    + "_iac-support-email"  = "support@skyvafnir.cloud"
    + "cost_center"         = "marketing"
    + "environment"         = "sky-dev"
    + "instance"            = "demo"
    + "org_code"            = "sky"
    + "tier"                = "dev"
    + "owner"               = "gissur@skyvafnir.is"
    + "project"             = "sky-dev-demo"
  }
}
```

```
Plan: 1 to add, 0 to change, 0 to destroy.
```

```
# main.tf
```

```
locals {  
  tags = {  
    instance = "demo"  
    org_code = "sky"  
    tier      = "dev"  
    owner    = "gissur@skyvafnir.cloud"  
  }  
}
```

```
resource "azurerm_virtual_network" "this" {  
  name                = module.defaults.resource_name  
  resource_group_name = var.resource_group_info.name  
  location            = var.resource_group_info.location  
  address_space       = [var.virtual_network_cidr] # "10.0.0.0/16"  
  
  tags = module.defaults.tags  
}
```



# Af hverju Terraform?

- Rekstrartækt
- Áreiðanlegt
- Breytingastjórn
  - Rekjanleiki (*e. traceability*)
  - Endurtakanleiki (*e. repeatability*)
- Fellur vel að GitOps hugmyndafræði
- Endurnýtanlegt

# Af hverju Terraform?

- Ekki **BARA** AWS/Azure/GCP/etc.
- Risastórt vendor eco-system
- Risastórt samfélag

## Tier

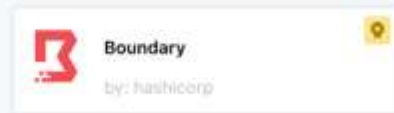
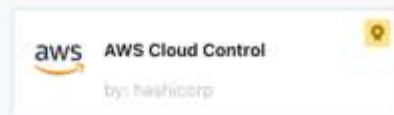
- Official
- Partner
- Community

## Category

- HashiCorp Platform
- Infrastructure Management
- Public Cloud
- Asset Management
- Cloud Automation
- Communication & Messaging
- Container Orchestration
- Continuous Integration/Deployment (CI/CD)
- Data Management
- Database
- Infrastructure (IaaS)
- Logging & Monitoring
- Networking
- Platform (PaaS)
- Security & Authentication
- Utility
- VCS (Version Control)
- Web Services

## Providers

Providers are a logical abstraction of an upstream API. They are responsible for understanding API interactions and exposing resources.



# Lykilatriði góðrar IaC vegferðar

# Lykilatriði góðrar IaC vegferðar

- Hafa skýra hönnun / arkitektúr.
  - Terraform er ekki töfrasproti
- Ekki finna upp hjólið:
  - Fylgja best-practices í uppbyggingu innviða, kóða og tengrða tólakeðja.
- Vera óhrædd við að ítra.

# Lykilatriði góðrar laC vegferðar

**Stuðningur (og skilningur)  
stjórnenda**

# Stuðningur frá stjórnendum

- IaC vinna er langtíma-fjárfesting.
- Kostnaður er front-loaded.
- Mikil samkeppni um IaC sérfræðinga.
- Sparnaður oft ósýnilegur:
  - Færri þjónustufrávik / útköll
  - Hraðari ítranir
  - Minna viðhald

**/ ský /**



**SKÝVAFNIR**

**TAKK**