


Infra :
Donnez de
l'autonomie à
vos
dével
avec



octoDNS

DEV**OX**TM France



N'avez vous jamais rêvé d'un
monde où ... 



**La création d'enregistrements
DNS est accessible à toutes et
tous? 🤔**



**Le DNS n'est plus réservé aux
Ops? 😇**



**Vous pouvez migrer de
l'on-prem' vers du Cloud en
moins de 5mn*? 🤯**

*sous réserve de l'effet démo ...





~ #whoami

Julien

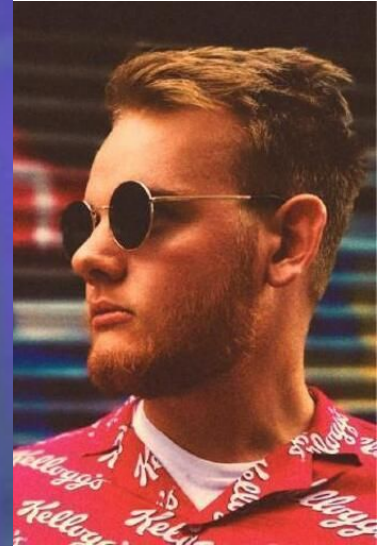
(ex) Sys

IT Man

Network

Auteur

#Network



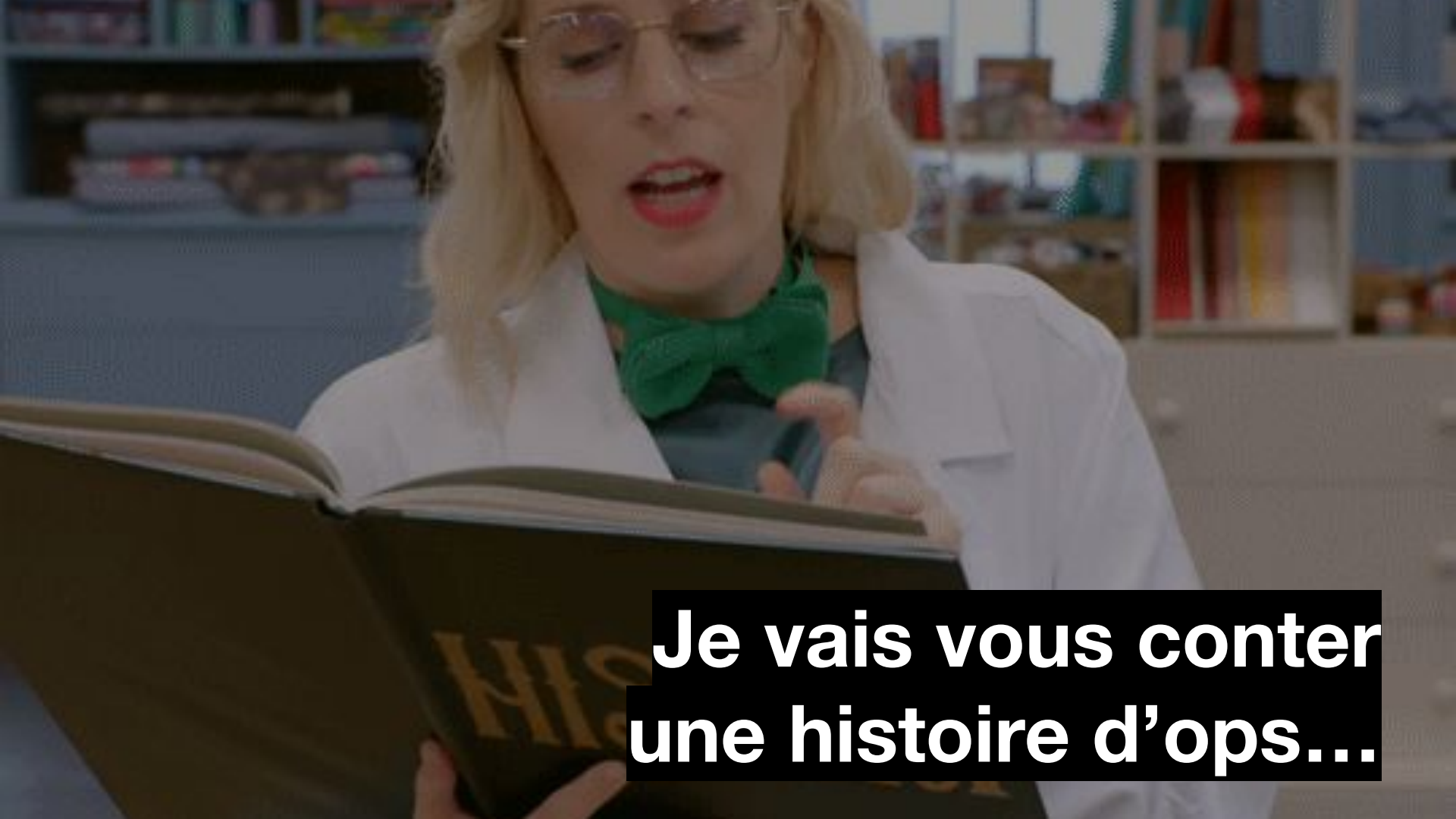
@ju_hnny5

C'est moi 😊



Et oui, je ne dors pas beaucoup... 😊





**Je vais vous conter
une histoire d'ops...**

Voici Michael 🙄



Des 90s à 2005 : L'âge d'or de Bind 🤠

Les balbutiements de l'automatisation (*gestion de configuration, pet vs cattle*)

Une administration via *Telnet/SSH* :

- Modification dans un fichier de zone
- Les utilisateurs doivent posséder un minimum de connaissances pour ne pas casser le fichier de zone ... 😬

N'expose pas d'API





L'arrivée de PowerDNS* : la modernité 🥰

Une API REST

Gestion simplifiée des zones

Une CLI performante

Un gros défaut : une API mono tenant.

*pdns_server (authoritative server)

Voici “Le Cloud”



Une solution écrasante

AWS Route53 🤖

Gain de temps de gestion des instances on-prem :

- Pas besoin de gérer la répartition de charge avec *dnsdist*

Possibilité de gestion “as-code” avec des outils comme **Terraform*** ou **Ansible*** par exemple.

*<https://www.terraform.io/>

*<https://www.ansible.com/>



Une solution écrasante : **AWS Route53** 🥰

Magic Quadrant de **Gartner**.

Figure 1. Magic Quadrant for Cloud Infrastructure and Platform Services



Source : <https://bit.ly/42U3GrZ>

Gérer ses enregistrements avec **Ansible** ?





Gérer ses enregistrements avec **Ansible** ?

- Le module Ansible Route53 **n'est pas idempotent** (enfin, pas totalement)...
- Il est lent quand on possède un très grand nombre de zones/enregistrements.
- Il attend rapidement les limites d'appels d'API Route 53 (environ 5 requêtes /sec).
- Pas de possibilité de migrer des enregistrements on-prem vers Route53 simplement (ou sans passer par scripting).



Gérer ses enregistrements avec **Terraform** ?



x2



Gérer ses enregistrements avec **Terraform** ?

- Tout comme **Ansible**, lent quand on possède un très grand nombre de zones/enregistrements.
- Format très (trop) verbeux ...

```
1 resource "aws_route53_record" "www-dev" {
2     zone_id = aws_route53_zone.primary.zone_id
3     name     = "www"
4     type     = "CNAME"
5     ttl      = 5
6
7     weighted_routing_policy {
8         weight = 10
9     }
10
11     set_identifier = "dev"
12     records       = ["dev.example.com"]
13 }
```



Gérer ses enregistrements avec Terraform ?

Idéal pour des petites zones (dans des projets restreints).

Un enfer quand on souhaite migrer d'un provider à un autre.

- Le format n'étant pas identique.
- Obligé de “scripter” pour transférer.

```
1 resource "aws_route53_record" "www-dev" {
2   zone_id = aws_route53_zone.primary.zone_id
3   name     = "www"
4   type     = "CNAME"
5   ttl      = 5
6
7   weighted_routing_policy {
8     weight = 10
9   }
10
11   set_identifier = "dev"
12   records       = ["dev.example.com"]
13 }
```

AWS Route53

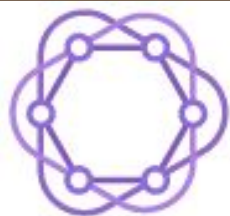
```
1 recordsets = [
2   {
3     name     = ""
4     type     = "NS"
5     ttl      = 300
6     records = [
7       "127.0.0.1",
8     ]
9   }
10 ]
```

GCP Cloud DNS



Gérer ses enrégistrement avec
l'interface Web

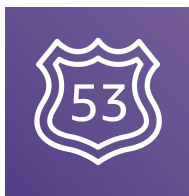




octoDNS

OctoDNS, késako ?

- Créé par **GitHub** pour gérer leur infrastructure DNS
- Première release (Licence MIT) en **2017**
- Écrit en **Python** (3) 🤔
- *Migrer rapidement d'un provider DNS à un autre (pratique en cas de panne). 🔥*
- Outil dit "stateful".
- Supporte plusieurs types d'entrées (statiques ou dynamiques) tout comme de destinations.



POWERDNS



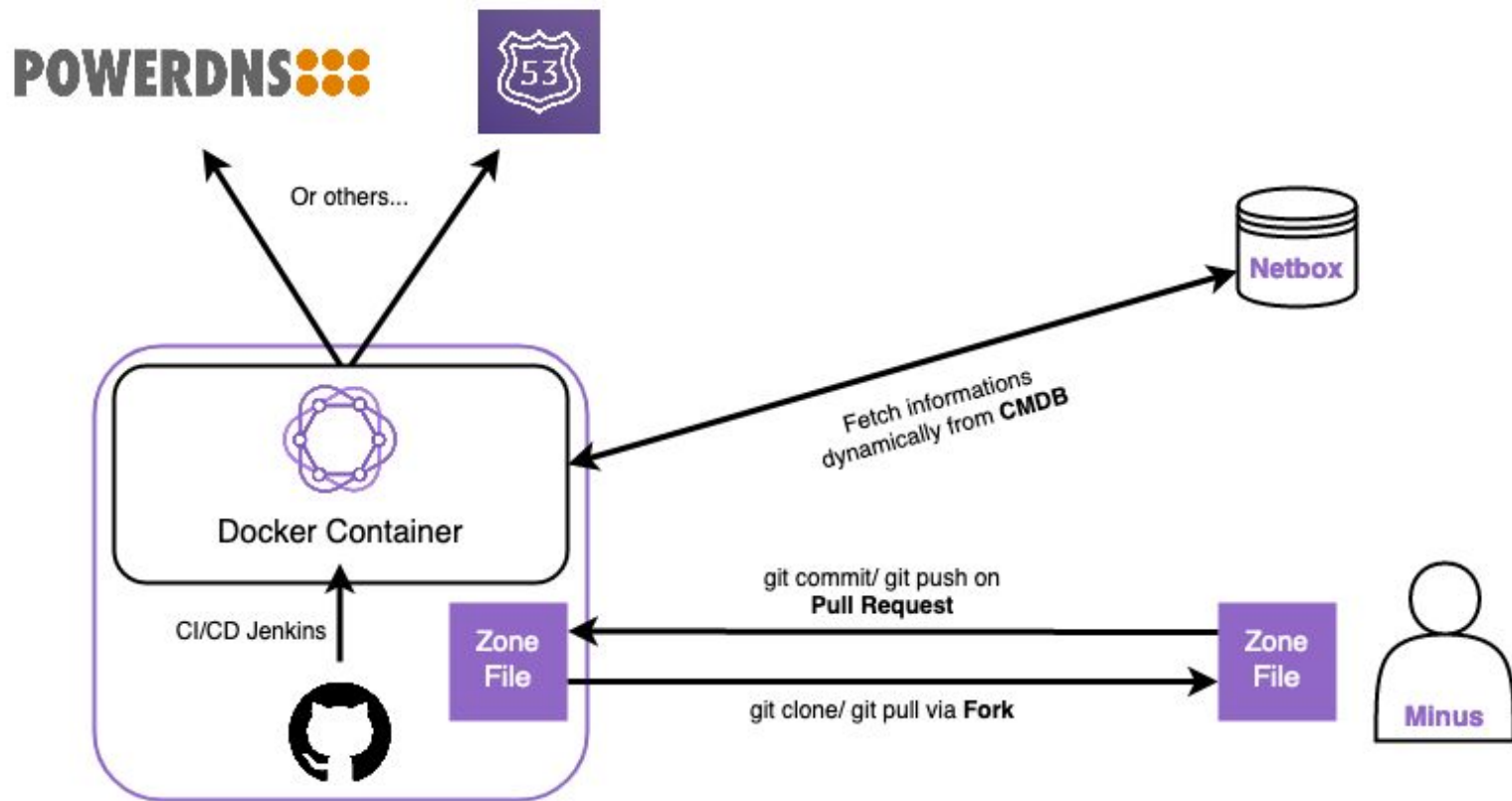
CLOUDFLARE



BIND

Berkeley Internet Domain Name

OctoDNS, késako ?



*Fonctionnement appliqué chez Deezer 😊

Pourquoi OctoDNS ?

**Parce qu'il fait les choses
simplement et de manière
agnostique. 😅**

Configuration explicite et accessible

- Au format **YAML**
- Définition des zones gérées
- Définition des sources et destination

```
1 ---
2 providers:
3   zone_records_yaml:
4     class: octodns.provider.yaml.YamlProvider
5     directory: zones/
6     default_ttl: 3600
7     enforce_order: False
8   route53:
9     class: octodns_route53.Route53Provider
10    access_key_id: env/AWS_ACCESS_KEY_ID
11    secret_access_key: env/AWS_SECRET_ACCESS_KEY
12    max_changes: 100
13
14 zones:
15   example.org.:
16     sources:
17       - zone_records_yaml
18     targets:
19       - route53
```



Création d'enregistrement accessible

- Utilisation du format **YAML** pour déclarer ses enregistrements.
- Exemple pour déclarer **foo.example.org** :

```
1 resource "aws_route53_record" "foo" {
2   zone_id = aws_route53_zone.primary.zone_id
3   name     = "foo.example.org"
4   type     = "A"
5   ttl      = 300
6   records  = ["10.0.0.100"]
7 }
```

Terraform



```
1 foo:
2   ttl: 300
3   type: A
4   value: 10.0.0.100
```

OctoDNS



Migrer/ récupérer facilement une zone 🤨

- Permet de migrer d'un provider à l'autre sans effort particulier via la commande `octodns-sync`.
 - Peu importe la ou les source(s) ou la ou les destination(s).
- Permet de récupérer le contenu d'une zone et de le convertir au format OctoDNS via la commande `octodns-dump`.

```
1 > octodns-dump --config-file=2demo/config/config.yml --output-dir=2demo/dump/ totozone.com. powerdns
2 2023-03-29T12:47:47 [8295121216] INFO Manager __init__: config_file=2demo/config/config.yml (octoDNS 0.9.21)
3 2023-03-29T12:47:47 [8295121216] INFO Manager _config_executor: max_workers=2
4 2023-03-29T12:47:47 [8295121216] INFO Manager _config_include_meta: include_meta=False
5 2023-03-29T12:47:47 [8295121216] INFO Manager __init__: global_processors=[]
6 2023-03-29T12:47:47 [8295121216] INFO Manager __init__: provider=powerdns (octodns_powerdns 0.0.3)
7 2023-03-29T12:47:47 [8295121216] INFO Manager dump: zone=totozone.com., output_dir=2demo/dump/, output_provider=None, lenient=False, split=False, sources=['powerdns']
8 2023-03-29T12:47:47 [8295121216] INFO Manager dump: using custom YamlProvider
9 2023-03-29T12:47:47 [8295121216] INFO PowerDnsProvider[powerdns] populate: found 1 records, exists=True
10 2023-03-29T12:47:47 [8295121216] INFO YamlProvider[dump] plan: desired=totozone.com.
11 2023-03-29T12:47:47 [8295121216] INFO YamlProvider[dump] plan: Creates=1, Updates=0, Deletes=0, Existing Records=0
12 2023-03-29T12:47:47 [8295121216] INFO YamlProvider[dump] apply: making 1 changes to totozone.com.
```



Ce qu'il ne fait pas :

- **Ne permet pas de gérer la configuration des serveurs DNS.**
 - Il n'est pas là pour remplacer des outils comme **Ansible**.
- **Ne permet pas de changer la configuration des registrar** (ex : Gandi).





Quoi ?!

Laisser les devs faire de l'ops ?

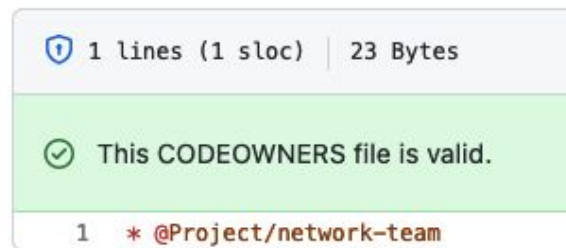




L'autonomie

L'autonomie ☐

- **Définition d'un cadre** (ex : dépôt Git avec workflow + CI/CD)
- **L'autonomie n'empêche pas le contrôle ! ☐**
 - L'équipe propriétaire du dépôt possède un droit de veto sur les modifications (ex : équipe réseau).
- Des **CODEOWNERS** pour plus d'efficacité. 🤖



L'autonomie

- **La communication** 🧐
 - Des channels dédiés
 - Chan dédié aux **questions portant sur le fonctionnement d'OctoDNS** ou sur un **besoin d'accompagnement**.
 - Ex : #discuss-octodns



L'autonomie

- **La communication** 🧐
 - Des channels dédiés
 - Chan dédié aux modifications sur le ou les dépôts OctoDNS (via un bot Github par exemple (et son template définit dans `.github/PULL_REQUEST_TEMPLATE.md`)).



OctoBot APPLI 15 h 33

[REDACTED] Pull request submitted by [jbrault](#)

#118 New subdomain creation

New subdomain creation [REDACTED]

Pull Request type

- Feature

Zone name

- [REDACTED]



L'autonomie

- **Gain :**

- Augmentation de la vélocité du développeur.
- Retire du toil aux équipes d'infrastructure/réseau (sauf la partie de review).
- Permet d'éviter les erreurs en passant par une application orchestrée par CI/CD des modifications.



L'autonomie

- **Gain :**


- Retour arrière facile (via `git revert`) en cas d'erreur (sur la zone même).
- Obtenir plus de contexte (avec les commits) sur les modifications apportées :
 - Permet également de suivre qui a fait une modification.
 - Permet de répondre à : “pourquoi cette modif” ?



Workflow



Workflow sur une Pull Request (PR)

1. **Nouvelle PR avec des enregistrements créée**
2. Validation de la syntaxe
3. Application sur l'environnement de *staging*
4. Report de l'état des enregistrements sur l'env de staging
5. **Pr validée** 



Workflow sur une Pull Request (PR)

1. Merge de la branche sur master
2. Application sur la production
3. Report de l'état des enregistrements sur l'env de prod
4. Enregistrements dispos !



A man with a beard, wearing a blue zip-up hoodie, is sitting at a desk and typing on a silver laptop. He has a wide-eyed, open-mouthed expression of surprise or excitement. The background is a plain, light-colored wall. The text "Demos !" is overlaid in the center of the image.

Demos !

Démo 1 : Créer ses premiers enregistrements. 



Démo 2 : **Récupérer les enregistrements d'une zone (dump).** 🐱💕



**Démo 3 : Migrer d'on-premise
vers le cloud. 🐱**



Instant pub ... □



<https://boutique.ed-diamond.com/en-kiosque/1652-linux-pratique-136.html>



A man with a mustache, wearing a red beret, a red neckerchief, and a black and white striped t-shirt, is shown from the chest up. He is looking slightly to his right. In the background, the Eiffel Tower is visible against a clear blue sky with a few wispy clouds. The word "Voilà" is written in white text to the right of the man.

Voilà`

Merci

DEVOXX™
France



A close-up shot of a man in a dark suit, light blue shirt, and patterned tie. He is looking down and slightly to his left with a faint smile. The background is an office with shelves and a computer monitor.

☐ **Merci pour votre attention !** ☐