
COURS COMPLET : DNS & AWS

Route 53

PARTIE 1 – LE DNS (Domain Name System)

◇ 1.1 – Définition du DNS

DNS (Domain Name System) est le système utilisé pour **traduire les noms de domaine** (ex : `google.com`) en **adresses IP** compréhensibles par les ordinateurs (ex : `142.250.190.14`).

Sans DNS, tu serais obligé de retenir des adresses IP pour chaque site.

◇ 1.2 – Objectif du DNS

- Faciliter la **navigation web** en utilisant des noms lisibles.
 - Assurer la **résolution d'un nom de domaine** vers son serveur hébergeant le site web.
 - Servir d'annuaire **distribué et hiérarchique** de l'Internet.
-

◇ 1.3 – Fonctionnement du DNS

Étapes de la résolution DNS

1. Tu tapes `example.com` dans ton navigateur.
 2. Le navigateur interroge le **DNS Resolver** local (souvent géré par ton FAI).
 3. S'il ne connaît pas la réponse, il interroge :
 - Le **serveur racine DNS**
 - Puis le **serveur TLD** (ex : `.com`)
 - Puis le **serveur de noms autoritaire** pour `example.com`
 4. Ce dernier retourne l'adresse IP.
 5. Ton navigateur s'y connecte.
-

◇ 1.4 – Types d'enregistrements DNS

Type	Description	Exemple
A	Nom → IP v4	A <code>hooyia.com</code> → <code>3.93.12.123</code>

Type	Description	Exemple
AAAA	Nom → IP v6	AAAA hooyia.com → ::1
CNAME	Alias vers un autre nom	CNAME www → hooyia.com
MX	Serveur email	MX 10 mail.hooyia.com
TXT	Texte libre (SPF, DKIM)	TXT "v=spf1 include:mailgun.org..."
NS	Serveurs DNS autoritaires	NS ns-123.awsdns.com
SOA	Infos sur la zone DNS	TTL, email admin, etc.
PTR	IP → nom (résolution inverse)	PTR 123.12.93.3.in-addr.arpa

◇ 1.5 – Outils pour tester le DNS

- nslookup example.com
 - dig example.com
 - <https://dnschecker.org>
 - <https://intodns.com>
-

☛ PARTIE 2 – AWS ROUTE 53

◇ 2.1 – Qu'est-ce que Route 53 ?

Amazon Route 53 est un service DNS managé de AWS.

🔗 Il permet :

- D'acheter et gérer des **noms de domaine**
 - De créer des **zones DNS**
 - De gérer des **enregistrements DNS**
 - De configurer du **routing intelligent** (latence, géolocalisation)
 - D'ajouter des **vérifications de santé (Health checks)**
-

◇ 2.2 – Fonctionnalités clés

Fonction	Description
Hosted Zone	Conteneur d'enregistrements DNS pour un domaine
Record Set	Enregistrement de type A, CNAME, etc.
Health Check	Vérifie l'état de santé d'une ressource
Routing Policy	Stratégie de routage DNS
Domain Registration	Achat de domaines sur AWS

◇ 2.3 – Types de Routing Policy

Type	Utilité	Exemple
Simple	Une seule IP ou CNAME	A → 13.24.25.30
Weighted	Pondération entre plusieurs cibles	70% USA, 30% Europe
Latency-based	Dirige vers le serveur le plus rapide	Route vers us-east-1
Geolocation	Selon la localisation de l'utilisateur	Afrique → serveur au Maroc
Failover	Basculer si le serveur tombe	Serveur principal + backup
Multi-value answer	Réponse multiple avec vérification de santé	3 IPs, si 1 tombe, exclue

◇ 2.4 – Exemple de configuration de domaine

🔗 Objectif : Lier un domaine à une instance EC2

1. Créer une instance EC2 (récupérer son IP publique)
 2. Aller sur Route 53 > Hosted Zones
 3. Créer une nouvelle **zone hébergée publique**
 4. Ajouter un enregistrement :
 - Type : A
 - Nom : @ ou hooyia.com
 - Valeur : IP EC2
 5. Ajouter un enregistrement :
 - Type : CNAME
 - Nom : www
 - Valeur : hooyia.com
 6. Si domaine acheté ailleurs (GoDaddy, Namecheap), mettre à jour les **nameservers (NS)** dans leur dashboard avec ceux fournis par AWS.
-

◇ 2.5 – Utilisation d'un enregistrement TXT (ex : vérification Google)

- Aller dans la zone DNS
 - Ajouter :
 - Type : TXT
 - Nom : @
 - Valeur : "google-site-verification=abc123..."
-

◇ 2.6 – Exemple avec SSL via ACM + Route 53 + CloudFront

1. Créer un certificat SSL dans **AWS Certificate Manager**
2. Ajouter le TXT de vérification dans Route 53

3. Créer un **distribution CloudFront** avec le certificat
4. Pointer votre nom de domaine vers CloudFront dans Route 53

◇ 2.7 – Test des enregistrements DNS

```
dig +short hooyia.com
nslookup hooyia.com
ping hooyia.com
```

□ PARTIE 3 – Cas pratique pour ton tutoriel

Scénario : Déployer un site Django ou Node.js avec un domaine personnalisé

1. Déploiement de l'app (EC2, Elastic Beanstalk, etc.)
2. Créer une **zone hébergée dans Route 53**
3. Ajouter un **A record** → IP serveur
4. Configurer un **CNAME** → www vers root domain
5. Vérifier le tout avec dig, ping
6. Bonus : ajout d'un certificat SSL avec ACM

SCHÉMA DE RÉSUMÉ

```
Navigateur
↓
Résolveur DNS (FAI)
↓
Serveur racine → Serveur TLD (.com) → Serveur autoritaire
↓
Adresse IP renvoyée
↓
Connexion au site web
```

QUESTIONS & EXERCICES

1. À quoi sert un enregistrement CNAME ?
2. Quelle est la différence entre NS et SOA ?
3. Comment fonctionne le routage "latency-based" ?
4. Crée un domaine personnalisé pour un projet Django avec Route 53.

RESSOURCES UTILES

- Docs AWS Route 53 : <https://docs.aws.amazon.com/route53/>
- Générateur dig : <https://toolbox.googleapps.com/apps/dig/>
- DNS Explained (fr) : <https://www.cloudflare.com/fr-fr/learning/dns/>
- Visualisation : <https://www.nslookup.io/>