TP04 – Dynamic DNS — Organisation par GRANDES PARTIES

PARTIE 1 — Préparation de la topologie (ddns.imn)

Objectif :

Créer une topologie avec :

- un **serveur DNS** : *osiris*
- un **serveur DHCP** : *isis*
- des clients (eno, owt, pc1, pc2…)

Ce que tu dois faire :

1. Copier la topologie du TP22 (`dns1.imn`)
2. Enregistrer sous `ddns.imn` dans :
   `~/SCR.3.2/TP04/`
3. Assigner :
   - **osiris** : serveur DNS
   - **isis** : serveur DHCP
   - Les autres machines : sans IP statique (elles utiliseront DHCP)

Pourquoi ?

Parce que :

- DHCP doit distribuer IP + DNS + nom de domaine,
- DHCP doit envoyer des mises à jour dynamiques au DNS.

PARTIE 2 — Préparation des fichiers DNS (osiris)

Objectif :

Créer les fichiers de zone que BIND pourra *modifier dynamiquement*.

Étapes :

(1) Copier les fichiers du TP22 dans TP04

Tu copies depuis ton dossier TP22 tout ce qui concerne DNS :

```
named.conf.local
db.tp.scr
db.1.168.192
```

Tu renommes :

```
osiris.named.conf.local
osiris.db.tp.scr
osiris.db.1.168.192
```

(2) Éditer les fichiers de zone

Le TP dit : garder seulement :

- les enregistrements pour **osiris**,
- ajouter ceux pour **isis**,

- supprimer toutes les autres machines (eno, owt, pc1, pc2…).

tp04.SCR.3.2

Parce que ces autres machines recevront leurs enregistrements **dynamiquement via DHCP**, donc pas dans les fichiers statiques.

(3) Déplacer les fichiers dans `/var/lib/bind/`

Très important : BIND **n'a pas le droit d'écrire** dans `/etc/bind/`.

Les fichiers de zones finalisés doivent être placés dans :

```
/var/lib/bind/db.tp.scr
/var/lib/bind/db.1.168.192
```

Pourquoi ?

- Parce que `named` tourne sous l'utilisateur **bind**,
- Il doit créer un fichier journal `.jnl` pour les mises à jour dynamiques,
- `/etc/bind` n'est pas modifiable par bind.

tp04.SCR.3.2

PARTIE 3 — Génération et configuration de la clé TSIG (sécurité)

Objectif :

Permettre à DHCP (isis) de mettre à jour DNS (osiris) en toute sécurité.

Générer la clé sur *osiris* :

```
tsig-keygen -a md5 tp.scr-key > ddns.key
```

Installer la clé :

Sur osiris :

```
/etc/bind/ddns.key
chmod 640 ddns.key
chown root:bind ddns.key
```

Sur isis :

```
/etc/dhcp/ddns.key
chmod 640 ddns.key
chown root:bind ddns.key
```

Pourquoi ?

TSIG = Transaction Signature
→ sécurité = le DNS n'accepte des updates **que du DHCP**, pas d'un intrus.

PARTIE 4 — Configuration DNS dynamique (osiris)

Objectif :

Dire au serveur DNS de permettre les mises à jour dynamiques venant de isis.

Modifier `/etc/bind/named.conf.local` (renommé ici en osiris.named.conf.local)

Exemple :

```
include "/etc/bind/ddns.key";

zone "tp.scr" IN {
    type master;
    file "/var/lib/bind/db.tp.scr";
    allow-update { key tp.scr-key; };
```

```
};

zone "1.168.192.in-addr.arpa" IN {
    type master;
    file "/var/lib/bind/db.1.168.192";
    allow-update { key tp.scr-key; };
};
```

Pourquoi ?

- `allow-update { key ... }` = autoriser DHCP uniquement.
- Le fichier de zone est dans `/var/lib/bind/` pour que BIND puisse écrire le `.jnl`.

PARTIE 5 — Configuration DHCP avec DDNS (isis)

Objectif :

Faire en sorte que DHCP :

- attribue des IP aux clients,
- génère automatiquement :
    - un enregistrement A (nom → IP)
    - un enregistrement PTR (IP → nom)
- envoie les mises à jour signées au DNS.

Modifier `/etc/dhcp/dhcpd.conf` (fichier isis.dhcpd.conf)

Exemple minimal :

```
include "/etc/dhcp/ddns.key";

ddns-update-style standard;
update-static-leases on;

option domain-name "tp.scr";
option domain-name-servers 192.168.1.158;   # osiris

zone tp.scr. {
    primary 192.168.1.158;
    key tp.scr-key;
}

zone 1.168.192.in-addr.arpa. {
    primary 192.168.1.158;
    key tp.scr-key;
}

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.11 192.168.1.200;
    option routers 192.168.1.254;
}
```

Pourquoi ?

Parce que DHCP doit :

- **savoir où envoyer** les mises à jour DNS → directives `zone {...}`;
- **les signer** → directive `key`;
- fournir l'IP du DNS aux clients → option `domain-name-servers`.

PARTIE 6 — Lancement manuel des services en mode debug

Objectif :

Observer les transactions DDNS en temps réel.

Sur osiris (DNS) :

```
named -u bind -g
```

Cela garde `named` en avant-plan et affiche les mises à jour.

Sur isis (DHCP) :

```
dhcpd -d
```

Cela permet de voir :

- DHCPDISCOVER
- DHCPOFFER
- DHCPREQUEST
- DHCPACK
- et surtout → la génération des messages DDNS

Pourquoi ne pas lancer via `service bind9 start` ?

Parce que tu ne verrais **aucun log**, et tu ne pourrais pas analyser les mises à jour dynamiques.
Le TP impose explicitement l'exécution en mode debug.

tp04.SCR.3.2

PARTIE 7 — Tests fonctionnels

1. Avant la requête DHCP

Sur un client (eno par exemple) :

```
ip a
cat /etc/resolv.conf
```

→ pas d'IPv4, pas de DNS.

---

2. Demande d'adresse :

```
dhclient -v eth0
```

Observations :

Côté DHCP (isis) :

- DHCPACK
- "Added new forward map ..."
- "Added reverse map ..."

Côté DNS (osiris) :

- Validation TSIG
- Écriture dans `.jnl`
- Mise à jour A + PTR

---

3. Test DNS :

```
dig eno.tp.scr
dig -x 192.168.1.X
```

Résultat attendu :

- le nom résout vers l'IP DHCP,
- l'IP résout vers le nom.

PARTIE 8 — Cas d'erreur : mauvaise clé TSIG

Le TP demande de modifier légèrement la clé du DHCP pour provoquer une erreur.

tp04.SCR.3.2

Symptômes :

DHCP (isis) :

→ *Continue à fonctionner normalement*.

DNS (osiris) :

→ Rejette les mises à jour :

```
tsig verify failure (BADSIG)
```

Test DNS :

```
dig eno.tp.scr
```

→ Réponse : **NXDOMAIN**

Pourquoi ?

Parce que la requête DDNS est :

- signée par DHCP,
- vérifiée par DNS.

Si la signature ne correspond pas, DNS rejette la mise à jour → donc pas d'enregistrement.

RÉSUMÉ PARFAIT DU TP04

Partie	Ce que tu fais	Pourquoi
1	Créer ddns.imn	Topo du TP
2	Préparer les zones	Utilisation dynamique
3	Créer clé TSIG	Sécurisation
4	Configurer DNS	Autoriser mises à jour
5	Configurer DHCP	Envoyer mises à jour
6	Lancer named & dhcpd en debug	Voir les transactions
7	Tester DHCP + DNS	Vérification du DDNS
8	Tester erreur de clé	Comprendre TSIG