Troubleshooting DNS

Situation 1

1) Outils utilisé & leur utilisation dans mon càs

- **Ifconfig :** Cette commande permet de voir quel sont toutes les caractéristiques réseau qu'un appareil. Elle permet par exemple de voir à quel IP l'appareil est connecté et s'il en a bien une. *Mon Utilisation :* j'ai utiliser cette commande pour vérifier si le client1 avait bien reçu une IP valide de la part du serveur DHCP.

```
root@client-1:/etc# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.13  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::e01c:ddff:fe1e:75b2  prefixlen 64  scopeid 0x20<link>
        ether e2:1c:dd:1e:75:b2  txqueuelen 1000  (Ethernet)
        RX packets 525  bytes 124844 (124.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 261  bytes 31624 (31.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- **Ping :** Cette commande a pour objectif d'analyser la connectivité d'un client au réseau. On peut utiliser la commande ping avec une adresse IP ou une URL et voir s'ils répondent. *Mon Utilisation :* J'ai utilisé ping pour voir la connective à internet au client, une première fois avec une IP qui a bien fonctionner et une seconde fois avec une URL qui à foiré.

```
root@client-1:/# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=2 ttl=112 time=45.1 ms
```

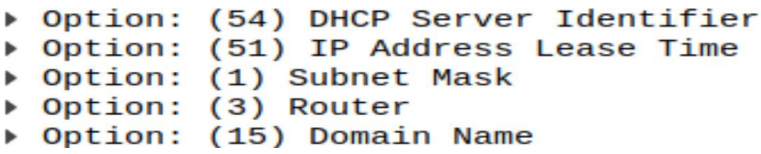
```
root@client-1:/# ping www.google.com
ping: www.google.com: Temporary failure in name resolution
```

- **dig :** Cette commande permet de vérifier si un serveur DNS fonctionne correctement avec une URL donnée en paramètre. *Mon Utilisation :* J'ai utilisé la commande dig sur le client1 avec l'IP du DNS pour vérifier si sa connectivité à internet fonctionnait avec la bonne IP du DNS. On peut voir dans le screen qu'il n'y a aucune erreur avec le : NOERROR

```
root@client-1:/# dig @192.168.0.1 www.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> @192.168.0.1 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 106
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a5e0fb2dd686f7a10100000065763f90ef4453322e3119a1 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         300     IN      A       142.251.36.36

;; Query time: 36 msec
;; SERVER: 192.168.0.1#53(192.168.0.1)
;; WHEN: Sun Dec 10 22:45:36 UTC 2023
;; MSG SIZE  rcvd: 87
```

- **Links :** Cette commande permet d'ouvrir un site web à partir d'une URL en ligne de commande. Cela confirme le bon fonctionnement de la connexion et s'assure que le serveur DNS remplit efficacement son rôle de conversion d'URL en adresse IP. *Mon Utilisation :* J'ai utiliser cette commande à la fin de ma correction de bug pour être sur que le client1 est connecter à internet.

- Wireshrark : cette application permet d'analyser les paquets qui sont envoyés sur le réseau et de lire leur contenu. Cela facilite la détection d'erreurs dans les protocoles applicatifs. *Mon Utilisation :* J'ai utiliser wireshark pour analyser les informations que fessait passer le serveur DHCP au client1 et remarquer qu'il y a pas l'adresse IP du serveur DNS.



2) Symptôme collecté

Après le lancement de la simulation ainsi que du service DHCP et DNS, nous pouvons effectuer un test avec le client1 pour vérifier s'il est effectivement connecté à Internet en utilisant la commande ping avec l'adresse IP 8.8.8.8 (www.google.com). Une fois cette étape réussie, nous procéderons à un nouveau test de la connectivité à Internet en utilisant le pigne avec une URL. Cela nécessitera que le client passe par le DNS et donc vérifier son fonctionnement. Malheureusement, cette tentative échoue, indiquant un problème lié au DNS. Afin de déterminer à quel DNS le client1 est connecté, nous pouvons consulter le fichier : *etc/resolv.conf*. En examinant ce fichier, il est notable qu'aucun DNS n'a été attribué, suggérant que le serveur DHCP n'a pas fourni cette information.

3) Explication du bug

Dans le but d'analyser ce problème lié au DNS, nous pouvons utiliser Wireshark pour vérifier si le serveur DHCP transmet toutes les informations réseau nécessaires lorsqu'un client fait une demande d'adresse IP donc l'IP du serveur DNS. Pour ce faire, nous devons lancer une capture du réseau avec Wireshrark et refaire une demande d'IP auprès du serveur DHCP (en utilisant la commande *dhclient -r | dhclient*). Après cela, en analysant les données capturées sur Wireshark, on peut effectivement constater que le serveur DHCP ne transmet aucune information concernant le DNS.

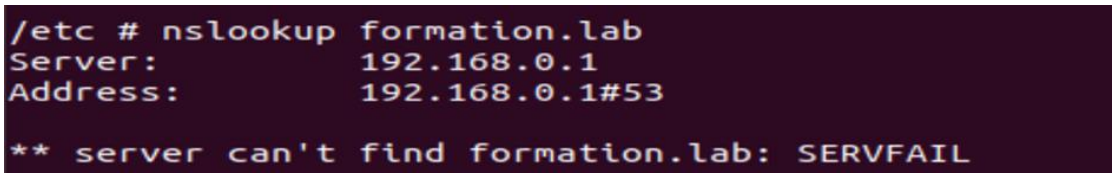
4) Proposition de solution

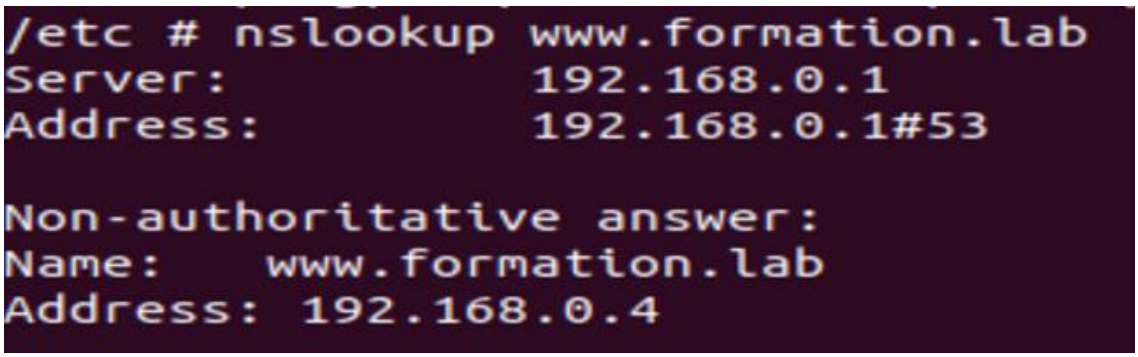
Pour résoudre ce problème où le serveur DHCP ne transmet pas l'IP du serveur DNS du réseau, il suffit d'ajouter la ligne suivante : *option domain-name-servers 192.168.0.1;* dans le fichier *etc/dhcp/dhcpd.conf*, au niveau des informations fournies par le serveur. Cette petite ligne de code permet de dire que l'IP du serveur DNS sur le réseau est 192.168.0.1 .

Situation 2

1) Outils utilisé & leur utilisation dans mon càs

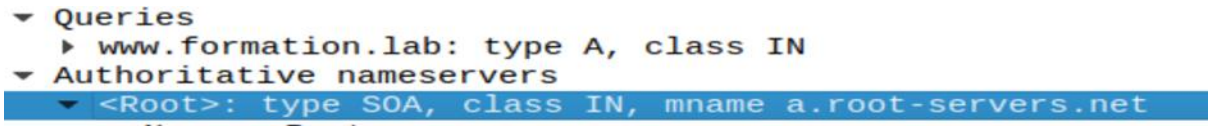
- Les anciennes commandes de la situation 1
- **nslookup :** Cette commande permet de récupérer l'IP d'un site à partir d'une URL et vice-versa. Elle est utile pour vérifier si le DNS parvient à trouver l'adresse IP associée à une URL. *Mon Utilisation :* J'ai employé cette commande sur le client pour vérifier si le serveur DNS pouvait correctement trouver l'adresse IP du site www.formation.lab. Initialement, il indiquait une impossibilité de trouver l'adresse IP, mais suite à ma correction de l'erreur, il était en mesure de le localiser correctement.





2) Symptôme collecté

Après le lancement de la simulation ainsi que des services DHCP, DNS et SOA, j'ai pu réaliser un test rapide avec le client1 pour confirmer sa connexion à Internet et le bon fonctionnement du serveur DNS en utilisant la commande ping avec l'adresse IP www.google.com. Ensuite, la cliente nous a informé qu'elle n'arrive plus à accéder au site sur l'intranet de l'entreprise. Pour vérifier cela, j'ai effectuer un ping vers le site www.formation.lab, et il semble y avoir effectivement un problème de résolution de cette URL spécifique. Pour examiner plus en détail les réponses du DNS, j'ai lancer une capture Wireshark sur le réseau et refaire un ping afin d'analyser les réponses du DNS. Après avoir effectué cette opération, il devient apparent que le DNS ne renvoie pas du tout l'adresse IP du serveur contenant l'intranet, mais quelque chose de totalement différent.



On peut aussi le voir avec la commande nslookup www.formation.lab que le serveru DNS ne répond pas avec l'adresse IP du server www de l'intranet :

```
/etc # nslookup formation.lab
Server:         192.168.0.1
Address:        192.168.0.1#53

** server can't find formation.lab: SERVFAIL
```

3) Explication du bug

Nous pouvons constater ici que le DNS ne nous répond pas du tout avec la bonne adresse IP, comme s'il n'avait pas été configuré pour la zone formation.lab. Une zone qui devrait renvoyer la requête vers le SOA qui dira quel adresse IP répondre.

4) Proposition de solution

Pour résoudre ce souci, on va devoir modifier la configuration du serveur DNS pour lui ajouter la zone formation.lab qui va rediriger la requête vers le serveur SOA, afin que celui-ci puisse bien traiter la demande et rendre la bonne adresse IP demandée. Le fichier à modifier est le *etc/bind/resolv.conf*, et nous devons y rajouter ce bout de code qui va indiquer que toute requête vers cette zone sera redirigée vers le SOA qui est en 192.168.0.2 :

```
zone "formation.lab" IN {
        type forward;
        forwarders { 192.168.0.2; };
        forward only;
    };
```

Une fois fait cela, on peut relancer le serveur DNS avec la commande named -g, pour ensuite refaire un ping avec le client vers l'intranet :

```
/etc # ping www.formation.lab
PING www.formation.lab (192.168.0.4) 56(84) bytes of data.
64 bytes from 192.168.0.4 (192.168.0.4): icmp_seq=1 ttl=64 time=0.358 ms
64 bytes from 192.168.0.4 (192.168.0.4): icmp_seq=2 ttl=64 time=0.759 ms
```

On peut donc réanalyser la requête avec Wireshrark, et on peut donc voir que le DNS répond bel et bien avec l'IP de sur serveur www de l'intranet.

```
▾ Queries
  ▾ www.formation.lab: type A, class IN
      Name: www.formation.lab
      [Name Length: 17]
      [Label Count: 3]
      Type: A (Host Address) (1)
      Class: IN (0x0001)
▾ Answers
  ▾ www.formation.lab: type A, class IN, addr 192.168.0.4
      Name: www.formation.lab
```

Dans tout cela, il ne faut pas oublier de lancer le serveur SOA avec la commande *named -g* pour qu'il puisse répondre avec la bonne adresse IP de www au client. Et aussi, une dernière chose, pour que le client puisse effectivement se connecter au site www.formation.lab de l'intranet, il faut également lancer le serveur www et le configurer comme indiqué dans le TP9 du cours.

Situation 3

1) Outils utilisé & leur utilisation dans mon càs:

- Les anciennes commandes de la situation 1 & 2

2) Symptôme collecté

Après le lancement de la simulation ainsi que des services DHCP et DNS, en essayant de lancer le serveur SOA, j'ai constaté que la zone formation.lab n'a pas pu se démarrer correctement.

```
11-Dec-2023 01:48:18.373 zone formation.lab/IN: NS 'soa.formation.lab.formation
.lab' has no address records (A or AAAA)
11-Dec-2023 01:48:18.373 zone formation.lab/IN: not loaded due to errors.
```

3) Explication du bug & Proposition de solution

Suite à ce constat, je suis allé voir dans le fichier contenant la zone dans *etc/bind/formation.lab*. Après une rapide comparaison entre les fichiers SOA de nos TP qui fonctionnaient bien et ceux de ce dépannage, j'ai remarqué qu'il y avait quelques erreurs. Il manquait le point après le nom de domaine. Ce point sert à indiquer à la zone que le domaine est terminé. La deuxième

correction concerne la ligne *@ IN MX 10 mail*, qu'il faut changer en *@ IN MX 10 mail.formation.lab.* . Il est nécessaire ici de préciser le nom de domaine.

```
$ORIGIN formation.lab.
$TTL 1d

@       IN      SOA     soa.formation.lab. vvds.ephec.be. (
                2001062501 ; serial
                21600      ; refresh after 6 hours
                3600       ; retry after 1 hour
                604800     ; expire after 1 week
                86400 )    ; minimum TTL of 1 day
;

@       IN      NS      soa.formation.lab.
@       IN      MX      10      mail.formation.lab.

soa             IN      A       192.168.0.2
resolver        IN      A       192.168.0.1
dhcpd           IN      A       192.168.0.3
www             IN      A       192.168.0.4
mail            IN      A       192.168.0.5
```

Après ce changement on peut relancer le SOA sans erreur et tester avec le client de ping par exemple www.formation.lab pour confirmer qu'il arrive bel et bien à se connecter à l'intranet:

```
PING www.formation.lab (192.168.0.4) 56(84) bytes of data.
64 bytes from www.formation.lab (192.168.0.4): icmp_seq=1 ttl=64 time=0.359 ms
64 bytes from www.formation.lab (192.168.0.4): icmp_seq=2 ttl=64 time=0.515 ms
```

On peut aussi verifier si le SOA rend bel et bien la bonne adresse IP à l'URL www.formation.lab qui est 192.168.0.4 .

```
root@client-1:/# nslookup www.formation.lab
Server:         192.168.0.1
Address:        192.168.0.1#53

Non-authoritative answer:
Name:   www.formation.lab
Address: 192.168.0.4
```

Et aussi, une dernière chose, pour que le client puisse effectivement se connecter au site www.formation.lab de l'intranet, il faut également lancer le serveur www et le configurer comme indiqué dans le TP9 du cours.