# Actividad DNS

## resolver.py

Este archivo contiene el código principal para ejecutar el resolver, para ejecutarlo se debe escribir en la terminal:

```
$ python resolver.py // o python3 dependiendo de las propias configuraciones
```

Con respecto al código presente en este archivo, primero se crea el socket en el cual se recibirán las querys por parte de los clientes, luego se crea el socket con el cual se comunicará el Resolver y los Name Servers, teniendo en cuenta que para este último socket la dirección debe tener al puerto 53. Por último, se envía la query resuelta con la IP del nombre de dominio consultado al cliente.

## utils.py

Este archivo posee funciones que son de utilidad para recibir, manejar y parsear los mensajes DNS. Dado que solamente posee funciones, las cuales se encuentran bien documentadas, no es necesario explayarse en detalle en la descripción de cada una de estas.

## Experimentación

A continuación se presentan los resultados y respuestas a los experimentos enunciados en la respectiva sección de la actividad:

- *Intente resolver el siguiente dominio con su programa* ***www.webofscience.com*** *¿Resuelve su programa este dominio? ¿Qué sucede? ¿Por qué? ¿Cómo arreglaría usted este problema?*

  El programa resuelve parcialmente este dominio, este entrega en la sección respuesta un RR de tipo CNAME lo cual indica un alias que apunta a otro nombre de dominio, en este caso **iwww.www.webofscience.com.akadns.net.**, y debido a la forma en que se encuentra programado el resolver (ignora toda respuesta que no sea de tipo A), considera esta como una respuesta[1] y por tanto retorna al cliente la query con esta única respuesta como se observa en la Figura 1, adicionalmente guarda en cache el dominio consultado en conjunto con la "dirección" IP de respuesta. Luego, en caso de volver a consultar por el dominio **www.webofscience.com** se retornará el campo RDATA de la respuesta como si fuera una dirección IP del dominio consultado lo cual producirá que el programa falle pues no podría enviar la query a esta "dirección". Este problema se puede arreglar verificando el tipo de respuesta recibida, en caso de que sea de tipo CNAME se debe realizar de forma iterativa consultas al nombre de dominio al cual apunta hasta obtener una respuesta de tipo A, así se obtendría la misma respuesta dada por el resolver Google mostrada en la Figura 2.

- *Ejecute el comando* `dig -p8000 @localhost www.cc4303.bachmann.cl` *¿Qué ocurre? ¿Qué habría esperado que ocurriera? Anote sus observaciones en su informe. Contraste sus observaciones con la respuesta de ejecutar* `dig @8.8.8.8 www.cc4303.bachmann.cl` *y utilice sus conocimientos sobre DNS para explicar por qué ocurre esto.*

[1] La función que detecta una respuesta `has_typeA()` retornará -1 como índice y como la lista de RRs de la sección Answer solo tiene un elemento, al buscar por el índice retornará este RR considerándolo erróneamente como si fuese de tipo A.

Al ejecutar el comando dado el programa falla y termina su ejecución pues encuentra un error a la hora de accesar a un RR inexistente en la sección Answer de la query luego de sucesivas consultas a Name Servers como se observa en la Figura 3, esto ocurre pues el resolver está programado de manera que solo considera RRs de tipo NS para el caso de la sección Authority y RRs de tipo A para la sección Additional, en este caso el resolver primero realiza la consulta en el único RR de la sección Additional, luego consulta al único RR de la sección Authority produciendo una sucesión de consultas que terminan con el programa fallando. Esto ocurre por dos razones, como se mencionó el resolver solo considera ciertos tipos de RRs, a la hora de obtener el RR de Authority este es de tipo SOA y dada la forma en que funciona `has_typeNS()` esta nos retorna este RR el cual incorrectamente se está considerando de tipo NS, adicionalmente el resolver falla pues **www.cc4303.bachmann.cl** no se encuentra escrito correctamente, **cc4303.bachmann.cl** no posee como subdominio a **www.**, por tanto cualquier resolver no encontrará la dirección IP de esta consulta tal y como se observa al consultar al resolver Google por esta dirección en la Figura 4.

- *Realice varias consultas a un mismo dominio y a través del modo debug vea a qué Name Servers y direcciones IP le pregunta su resolver en cada consulta. ¿Son siempre los mismos Name Servers? ¿Por qué cree usted que sucede esto?*

  Aplicando varias consultas sobre el mismo dominio, en este caso **.uchile**, en especifico a **eol.uchile.cl**, **mi.uchile.cl**, **ucampus.uchile.cl** y **uchile.cl** se observa que el resolver consulta en todas las situaciones sobre los mismos Name Servers y direcciones IP; luego de consultar sobre la raíz consulta al Name Server `cl2-tld.d-zone.ca.` con dirección IP 185.159.198.56 para posteriormente consultar al Name Server `ns1.uchile.cl.` con dirección IP 200.89.70.3 con el cual se obtiene la respuesta a la query principal. Este comportamiento sucede pues los servidores DNS a los cuales el resolver consulta son los mas cercanos a este, adicionalmente puede caber la posibilidad de que solamente este servidor conozca el dominio y por tanto desde cualquier parte del mundo el resolver consultaría a este servidor para este caso de dominio en particular.

```
✘ nahuel dig -p8000 @localhost www.webofsccience.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.1-Ubuntu <<>> -p8000 @localhost www.webofsccience.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34374
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.webofsccience.com.         IN      A

;; ANSWER SECTION:
www.webofsccience.com.  600     IN      A       104.247.82.172

;; Query time: 788 msec
;; SERVER: 127.0.0.1#8000(localhost) (UDP)
;; WHEN: Mon Apr 17 20:24:46 -04 2023
;; MSG SIZE  rcvd: 76
```

Figure 1: Respuesta de `resolver.py` al consultar **www.webofscience.com**

```
nahuel dig @8.8.8.8 www.webofscience.com

; <<>> DiG 9.18.12-0ubuntu0.22.04.1-Ubuntu <<>> @8.8.8.8 www.webofscience.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39581
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.webofscience.com.          IN      A

;; ANSWER SECTION:
www.webofscience.com.   30      IN      CNAME   iwww.www.webofscience.com.akadns.net
.
iwww.www.webofscience.com.akadns.net. 300 IN CNAME www.webofscience.com.edgekey.net.
www.webofscience.com.edgekey.net. 21600 IN CNAME e90971.dsca.akamaiedge.net.
e90971.dsca.akamaiedge.net. 20  IN      A       23.44.231.48
e90971.dsca.akamaiedge.net. 20  IN      A       23.44.231.16

;; Query time: 632 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Mon Apr 17 20:25:04 -04 2023
;; MSG SIZE  rcvd: 211
```

Figure 2: Respuesta del resolver Google al consultar **www.webofscience.com**

```
[NEW QUERY] (debug) Recibiendo query desde IP: ('127.0.0.1', 60555)
<<------------------------------>>
<<------------------------------>>
(debug) Consultando 'www.cc4303.bachmann.cl.' a '.' con direccion IP '192.33.4.12'
<<------------------------------>>
<<------------------------------>>
(debug) Consultando 'www.cc4303.bachmann.cl.' a 'cl2-tld.d-zone.ca.' con direccion IP '185.159.198.56'
<<------------------------------>>
<<------------------------------>>
(debug) Consultando 'ns1.digitalocean.com.' a 'ns1.digitalocean.com.' con direccion IP '192.33.4.12'
<<------------------------------>>
<<------------------------------>>
(debug) Consultando 'ns1.digitalocean.com.' a 'm.gtld-servers.net.' con direccion IP '192.55.83.30'
<<------------------------------>>
<<------------------------------>>
(debug) Consultando 'ns1.digitalocean.com.' a 'kim.ns.cloudflare.com.' con direccion IP '108.162.192.126'
<<------------------------------>>
Traceback (most recent call last):
  File "/home/nahuel/Documents/fcfm/9no-semestre/redes/actividades/resolver_dns/resolver.py", line 16, in <module>
    response = resolver(query_msg, resolver_socket)  # Procesamos la query con el resolver
  File "/home/nahuel/Documents/fcfm/9no-semestre/redes/actividades/resolver_dns/utils.py", line 155, in resolver
    rr = parse_dns_msg(response)['answer']['resource_records_list'][0]
IndexError: list index out of range
```

Figure 3: "Debug" de ejecución de `resolver.py` al consultar **www.cc4303.bachmann.cl**

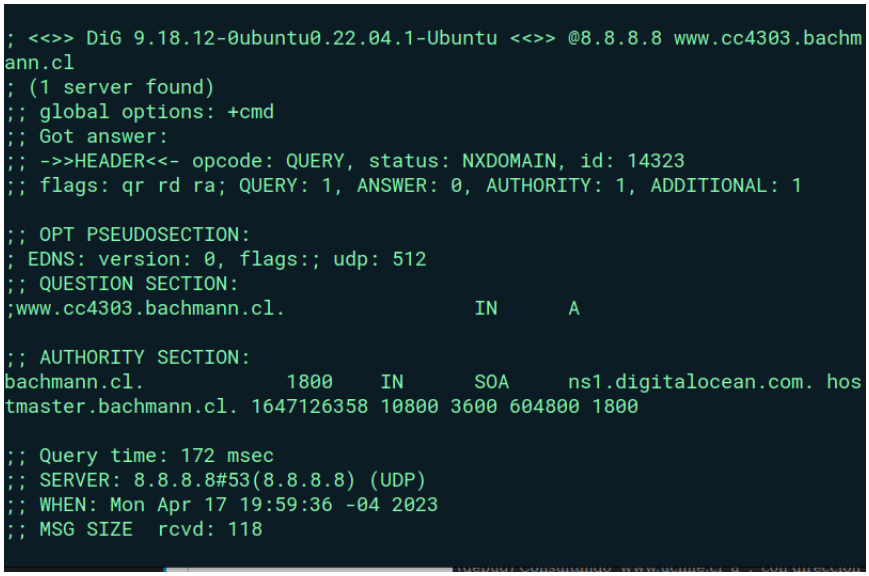


Figure 4: Respuesta del resolver Google al consultar **www.cc4303.bachmann.cl**