

3.6 ¿IP dinámica? No hay problema. DuckDNS

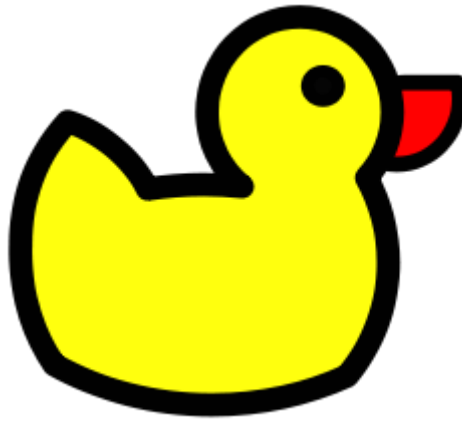


Imagen obtenida de <https://www.duckdns.org/>

Esta herramienta sirve para...

facilitarnos el acceder a nuestra red de casa desde fuera de la misma.

Vamos a aclarar la afirmación anterior. Nuestro router (que conecta la red de nuestro domicilio con la red del exterior) tiene una IP en esa red exterior que va cambiando con el tiempo salvo que tengamos contratado con nuestro ISP el servicio de IP fija. Dado que aprenderse una IP es complicado y dado que ese dato cambia con el tiempo el servicio de duckdns es muy interesante pues nos permite "ponerle nombre" a nuestra IP ([DNS](#)) ya que es más sencillo aprenderse pbloruizsoria.duckdns.org que 148.3.110.57 que, como hemos dicho, cambia con el tiempo. Este servicio no solo va a "poner nombre" a nuestra IP sino que va a hacer que cuando nuestra IP cambie el nombre pase a apuntar al nuevo valor de la IP.

Estaríamos hablando de un servicio [DDNS](#) (DNS dinámico)

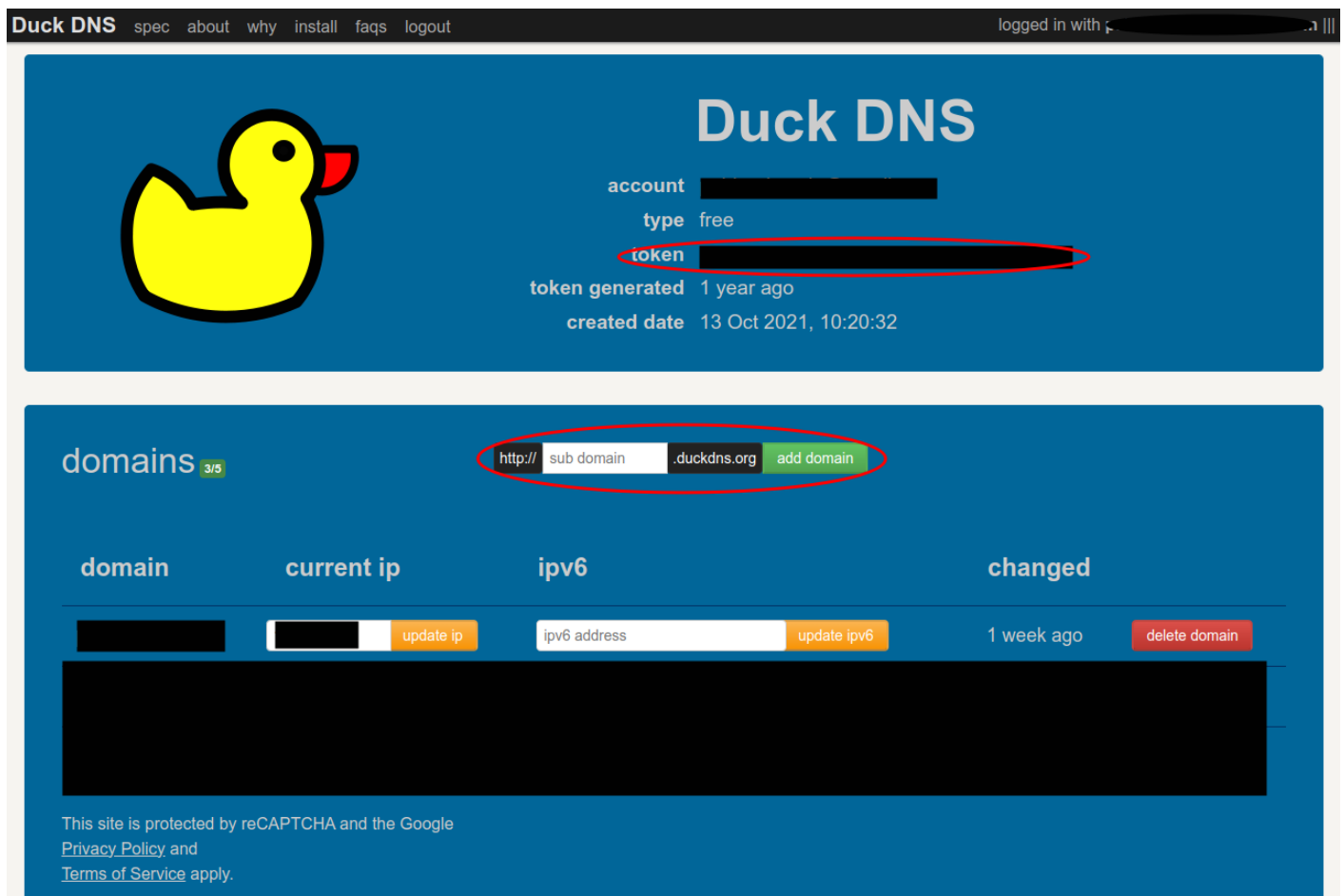
Web de proyecto y otros enlaces de interés

Web de duck DNS: <https://www.duckdns.org/>

Repositorio que usaremos para instalar este servicio: <https://github.com/linuxserver/docker-duckdns>

Despliegue

Primero deberemos acceder a la web <https://www.duckdns.org/> aquí crearemos un subdominio con el nombre que nos interese. De esta web nos interesará tanto el nombre del subdominio como el campo token que aparece en pantalla.



Duck DNS spec about why install faqs logout logged in with [redacted]

Duck DNS

account [redacted]
type free
token [redacted]
token generated 1 year ago
created date 13 Oct 2021, 10:20:32

domains 3/5

http:// [redacted] sub domain .duckdns.org add domain

domain	current ip	ipv6	changed
[redacted]	[redacted] update ip	ipv6 address update ipv6	1 week ago delete domain

This site is protected by reCAPTCHA and the Google Privacy Policy and Terms of Service apply.

Elaboración propia

Después, como con servicios anteriores, accedemos a la terminal y escribimos

```
cd $HOME
mkdir duckdns
cd duckdns
nano docker-compose.yml
```

Dentro del fichero escribimos el siguiente contenido

```
version: "2.1"
services:
  duckdns:
    image: lscr.io/linuxserver/duckdns:latest
    container_name: duckdns
    environment:
      - PUID=1000 #optional
      - PGID=1000 #optional
      - TZ=Etc/UTC #optional
      - SUBDOMAINS=subdomain1_que_hayas_configurado,subdomain2_que_hayas_configurado
      - TOKEN=token_que_aparece_en_la_web_duckdns_mira_mi_imagen
      - LOG_FILE=false #optional
    volumes:
      - /path/to/appdata/config:/config #optional
    restart: unless-stopped
```

Para salir del fichero pulsaremos `control + x` y guardaremos los cambios. Posteriormente ponemos en marcha los contenedores con `docker compose up -d` Aparecerá en pantalla algo similar a



```
pablo@raspberrypicatedu:~/duckdns $ docker compose up -d
[+] Running 8/8
✓ duckdns 7 layers [|||||] 0B/0B Pulled
✓ 3d082242de3c Already exists
✓ b81e0d862760 Already exists
✓ dda0d1ffe94b Already exists
✓ 2fe21492ca45 Already exists
✓ 361dc22a54c3 Already exists
✓ 83680b942bd5 Pull complete
✓ da5a66c234e7 Pull complete
[+] Running 2/2
✓ Network duckdns_default Created 0.2s
✓ Container duckdns Started 1.2s
```

Elaboración propia

Funcionamiento

Si ahora accedéis a <https://www.duckdns.org/> veréis que en el subdominio que acabáis de crear tiene un valor en current ip y que se ha actualizado hace poco.

Revision #7

Created 4 February 2023 10:05:58 by Pablo Ruiz

Updated 20 July 2023 17:29:50 by Pablo Ruiz