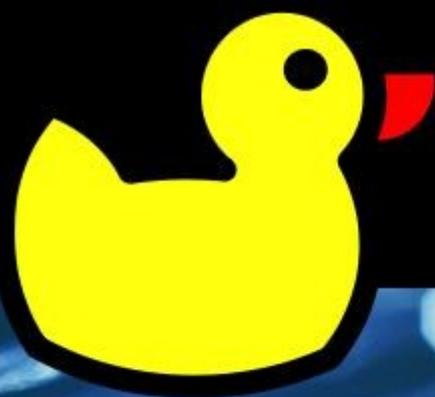


Como obtener una URL o dominio propio, gratis y con dirección IP Dinámica



DuckDNS

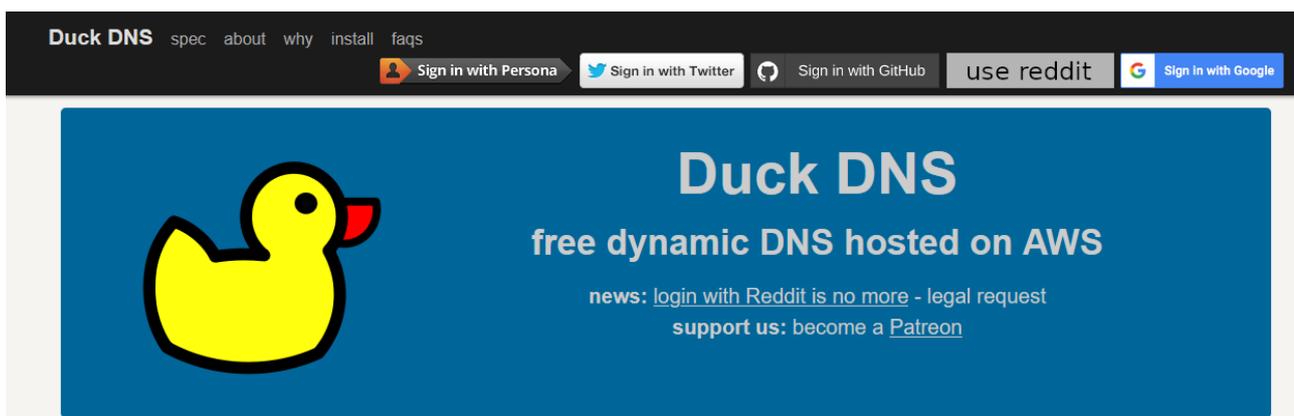
Como obtener una URL o dominio propio, gratis y con dirección IP Dinámica

Si queremos montar en nuestra casa un servicio en internet como un servidor Web, un servidor de VPN ... nos encontraremos, normalmente, con el problema de que nuestro router no tiene asignada una dirección IP fija, cambiando cada vez que reiniciamos el router. Si compramos un dominio, en el servidor DNS debemos indicar cual es la dirección IP a la que será traducida la dirección URL que suministremos, por ejemplo, en el navegador. La primera solución es contactar con nuestro ISP y contratar una dirección IP fija lo que nos supondrá un coste añadido.

Hay en Internet varios proveedores de dominios que admiten IP Dinámicas, para ello se ha de hacer periódicamente la actualización de la dirección IP dinámica de nuestro router en sus servidores DNS mediante la ejecución de forma periódica de un comando de actualización. Los hay de pago, pero también los hay gratuitos como el que vamos a ver, con la pega de que el nombre de dominio no es completo sino un subdominio de su dominio.

En DuckDNS, tras hacer login con una cuenta de Google, por ejemplo, nos van a permitir crear un subdominio de nuestra elección en su dominio **duckdns.org** de una forma muy sencilla. Una vez creado nos comunicarán un **token** que acompañaremos a la petición de cambio de IP cada vez que queramos que se realice.

Vamos a ver lo sencillo que es. Accedemos a <https://www.duckdns.org/>



Si tenemos cuenta en **Gmail** hacemos clic en **Sign in with Google**

Duck DNS spec about why install faqs logout logged in with [redacted]@gmail.com |||



Duck DNS

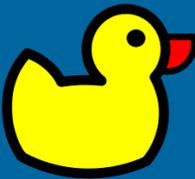
account [redacted]@gmail.com
 type free
 token [redacted]
 token generated 1 hour ago
 created date 16 Dec 2022, 10:28:26

domains 0/5 .duckdns.org

domain	current ip	ipv6	changed
This site is protected by reCAPTCHA and the Google Privacy Policy and Terms of Service apply.			

Ya podemos añadir un nuevo subdominio escribiendo su nombre en la caja de texto correspondiente y pulsando **add domain**. Si el nombre no está ya registrado tendremos un nuevo nombre de dominio para nuestro uso. Veremos la dirección IP actual a la que estará asociado el dominio que se corresponderá con la dirección IP de nuestro router. Esto es todo, mientras no cambie la dirección IP del router, funcionará perfectamente. De forma manual podemos volver a entrar en el mismo sitio y escribir la nueva dirección IP si es que no aparece ya y pulsar en **update ip**

Duck DNS spec about why install faqs logout logged in with [redacted]@gmail.com |||



Duck DNS

account [redacted]@gmail.com
 type free
 token [redacted]
 token generated 2 minutes ago
 created date 16 Dec 2022, 10:28:26

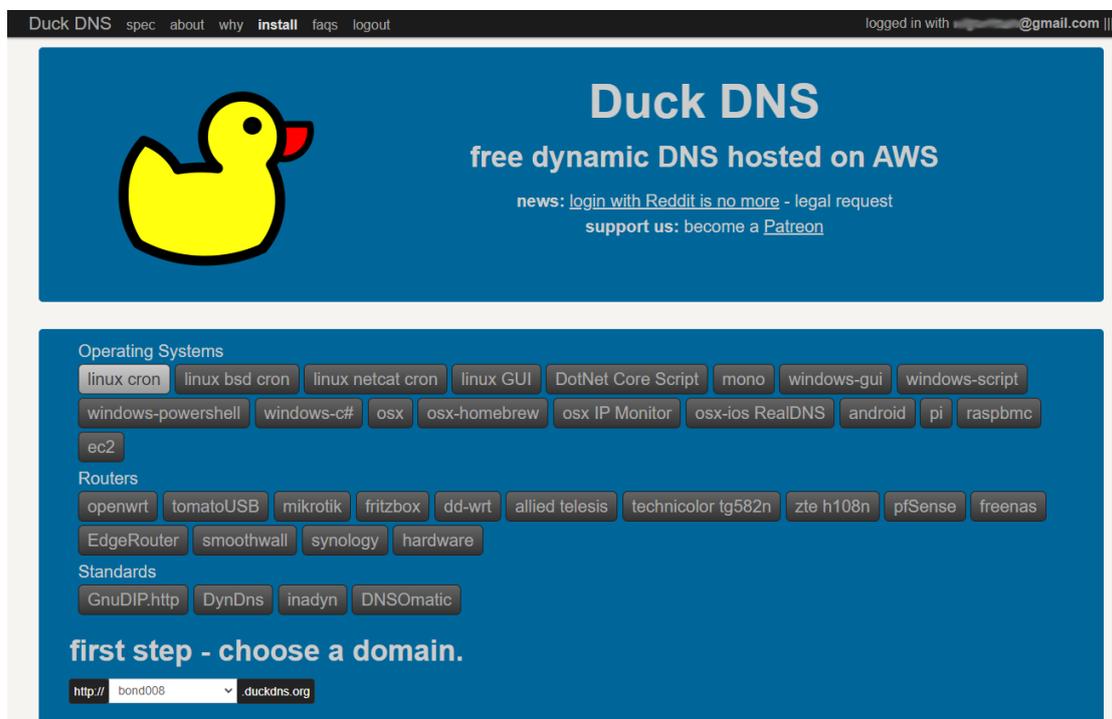
success: domain **bond008.duckdns.org** added to your account

domains 1/5 .duckdns.org

domain	current ip	ipv6	changed
bond008	<input type="text" value="current ip"/> <input type="button" value="update ip"/>	<input type="text" value="ipv6 address"/> <input type="button" value="update ipv6"/>	0 seconds ago <input type="button" value="delete domain"/>

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

Si en este mismo sitio web pulsamos en el menú superior en la opción **install** veremos una página en la que nos explican como hacer que la actualización de la dirección IP de nuestro router se haga de forma automática.



En ella seleccionamos el sistema operativo de la máquina que se va a encargar de ejecutar el comando de actualización, y el dominio que queremos actualizar. En la parte inferior de la página aparecerán las instrucciones:

first step - choose a domain.

http://

linux cron

if your linux install is running a crontab, then you can use a cron job to keep updated we can see this with

```
ps -ef | grep cr[on]
```

if this returns nothing - then go and read up how to install cron for your distribution of linux. also confirm that you have curl installed, test this by attempting to run curl

```
curl
```

if this returns a command not found like error - then find out how to install curl for your distribution. otherwise lets get started and make a directory to put your files in, move into it and make our main script

```
mkdir duckdns
cd duckdns
vi duck.sh
```

now copy this text and put it into the file (in vi you hit the i key to insert, ESC then u to undo) The example below is for the domain bond008
if you want the configuration for a different domain, use the drop down box above
you can pass a comma separated (no spaces) list of domains
you can if you need to hard code an IP (best not to - leave it blank and we detect your remote ip)
hit ESC then use use arrow keys to move the cursor x deletes, i puts you back into insert mode

```
echo url="https://www.duckdns.org/update?domains=bond008&token=bc0c700f-303-422c-800f-100c00000000&ip=" | curl -k -o ~/duckdns/duck.log -K -
```

now save the file (in vi hit ESC then :wq! then ENTER)
this script will make a https request and log the output in the file duck.log
now make the duck.sh file executeable

```
chmod 700 duck.sh
```

next we will be using the cron process to make the script get run every 5 minutes

```
crontab -e
```

copy this text and paste it at the bottom of the crontab

```
*/5 * * * * ~/duckdns/duck.sh >/dev/null 2>&1
```

now save the file (CTRL+o then CTRL+x)
lets test the script

```
./duck.sh
```

this should simply return to a prompt
we can also see if the last attempt was successful (OK or bad KO)

```
cat duck.log
```

if it is KO check your Token and Domain are correct in the duck.sh script

Si hemos seguido el artículo **Instalación VPN en casa** sobre como tener una VPN en nuestra casa con una Raspberry PI, tendremos ya una máquina con un sistema operativo Linux que vamos a utilizar para actualizar la IP. Cuando se hizo la instalación de la VPN hubo un momento en el que se pedía cual era la dirección del servidor para ser accedido desde el exterior y había dos opciones: introducir una IP con el problema de que esta IP puede cambiar, o introducir una URL, aquí es donde pondremos nuestro nuevo nombre de subdominio creado en DuckDNS:

bond008.duckdns.org

Si estamos en la situación final del artículo, conectaremos mediante SHH con Bitvise a nuestra Raspberry PI abriendo una terminal de consola. Cambiamos a modo superusuario

sudo su -

```
raspberrypi - usuario@192.168.1.201:22 - Bitvise xterm - usuario@vpn: ~
usuario@vpn:~$ sudo su -
Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.
root@vpn:~#
```

Y seguimos las instrucciones de la página de DuckDNS:

El comando **curl** ya lo encontramos instalado en Raspberry PI OS, por lo tanto no hace falta instalarlo, al igual que el servicio de **cron**.

Creamos bajo **/root** que es donde estamos la carpeta **duckdns** y nos cambiamos a ella editando a continuación un nuevo archivo con nombre **duck.sh**

```
mkdir duckdns
cd duckdns
nano duck.sh
```

```
raspberrypi - usuario@192.168.1.201:22 - Bitvise xterm - usuario@vpn: ~
root@vpn:~# mkdir duckdns
root@vpn:~# cd duckdns/
root@vpn:~/duckdns# nano duck.sh
```

Dentro del archivo escribimos el comando que nos sugieren, copiar y pegar (cambiar ~ por **/root**)

```
echo url="https://www.duckdns.org/update?
domains=bond008&token=bc0c700f-a263-422c-820f-186ecccccccc&ip=" | curl
-k -o /root/duckdns/duck.log -K -
```

Y lo guardamos pulsando Ctrl+X y contestando **Sí** a la pregunta que nos hace nano. A continuación cambiamos los permisos del archivo creado para que solo lo pueda ejecutar **root**:

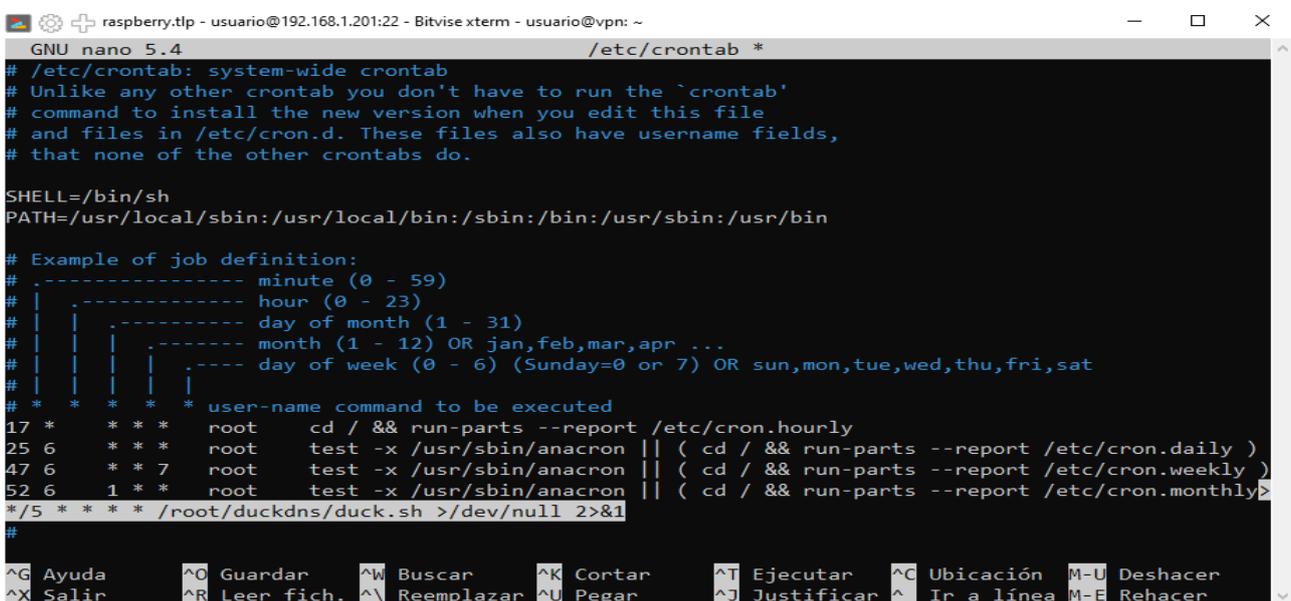
```
chmod 700 duck.sh
```

Editamos el archivo de configuración del programa programador de tareas **cron**:

```
nano /etc/crontab
```

Seleccionamos **nano** como editor de texto y añadimos la línea:

```
*/5 * * * * /root/duckdns/duck.sh >/dev/null 2>&1
```



```
GNU nano 5.4 /etc/crontab *
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

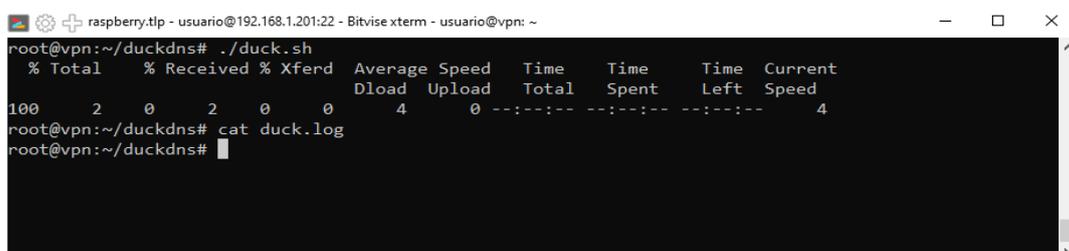
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# ----- minute (0 - 59)
# | ----- hour (0 - 23)
# | | ----- day of month (1 - 31)
# | | | ----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | ----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
*/5 * * * * /root/duckdns/duck.sh >/dev/null 2>&1
#
^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación M-U Deshacer
^X Salir ^R Leer fich. ^E Reemplazar ^U Pegar ^J Justificar ^_ Ir a línea M-E Rehacer
```

Guardamos y salimos con Ctrl+X y **Sí**. Probamos:

```
./duck.sh
```

Si todo ha ido bien tendremos un archivo **duck.log** con contenido vacío:



```
root@vpn:~/duckdns# ./duck.sh
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2 0 2 0 0 4 0 --:--:-- --:--:-- --:--:-- 4
root@vpn:~/duckdns# cat duck.log
root@vpn:~/duckdns#
```

Ya tenemos un nombre de dominio para nuestra IP dinámica.

Si compramos un dominio separado, por ejemplo, **bond008.net** y queremos que apunte también a nuestra IP dinámica no tenemos más que añadir un registro **CNAME** en el servidor DNS que tenga autoridad sobre **bond008.net** que apunte a **bond008.duckdns.org**. Normalmente cuando compramos un dominio nos ofrecen la gestión de los DNS del mismo a través de una aplicación web por lo que realizar esta tarea es cosa sencilla.