dnsmasq: DNS práctico y sencillo (con DHCP opcional)

Simon Kelley en su sitio web, y por cuenta propia, fue quien creó -y mantiene- a **dnsmasq** Así, sin mayúscula inicial (excepto si está al comienzo de una línea o después de un punto) es exactamente lo que su creador quiso que fuera: un programa sencillo, pero poderoso, que sirve como DNS sin mayores pretensiones, especialmente creado para redes de área local y con funciones adicionales, **veamos.**

Introducción

¿Qué hace un software hecho de manera artesanal y solitario, hecho por un británico en las tierras heladas del norte, sea tan especial? **Está bien hecho, es práctico, sencillo y directo,** *"la genialidad es la simplicidad"* dicen muchas personas. No subestimen a este pequeño programa porque es ampliamente utilizado para gestionar las direcciones IP de las máquinas virtuales basadas en **Docker** y manejado de manera masiva por **Kubernetes**. También es tan pequeño que está incrustado en millones de enrutadores inalámbricos y de ser necesario dispone de muchas características adicionales que se pueden instalar posteriormente:

- DHCP.
- Arranque de red con PXE, BOOTP y TFTP (si el DHCP está instalado y configuado).
- Acepta guiones en lenguaje LUA.
- Soporte para IPv6.
- Soporte para DNS seguros o DNSSEC.

Precisamente esta fama fue la que llamó la atención de los investigadores, los cuales se aplicaron a estudiarlo muy bien y en octubre de 2017 hallaron...

> **dnsmasq** tiene serias vulnerabilidades en la versión 2.77 y anteriores. Para que podamos usarlo debemos estar seguros de usar la versión 2.78 o superior; revisen con **apt show dnsmasq** el número de versión que les retribuye sus repositorios, de ser necesario descarguen desde su propia página web y lo compilan o en el caso de Ubuntu descargamos el paquete .deb del repositorio.

Al final, en una sección especial, colocamos todas las debilidades que adolecía dnsmasq, acá no las explicamos porque sería muy extenso. *El interés de los investigadores fue robustecer, fortalecer, asegurar a Docker y Kubernetes, bueyes de carga que llevan el peso de nuestros servidores en Internet.*

Esencialmente dnsmasq ofrece un DNS recursivo, es decir, si no sabe el dominio consultado entonces reenvía la consulta hacia el DNS configurado en la máquina GNU/Linux y una vez resuelto guarda en *memoria cache* dicha información, acelerando nuestra navegación. Además podremos colocar en el archivo **/etc/hosts** dominios adicionales, como por ejemplo anfitriones para servicios especiales como servidores FTP, de correo o Web. Si no queremos tocar ese archivo especial, también le podremos configurar que lea desde un archivo adicional los dominios que necesitemos declarar.

Instalando dnsmasq

Lo describiremos para **Ubuntu 18.04** pero todo esto se puede extrapolar para toda distribución basada en Debian.

systemd-resolve

Este señor **systemd-resolve** viene por defecto instalado y funcionando y utiliza la dirección 127.0.0.53 ***además de apoderarse del puerto 53, justo el que necesita dnsmasq para funcionar.***

De hecho systemd-resolved tiene dos métodos principales basados en API para "resolver" los dominios de las aplicaciones locales y a pesar de que el manual de dicho programa recomienda fuertemente que se utilicen primero estos dos métodos antes de usar el puerto 53 (*¿para qué entonces se adueñan de ese puerto??*). No explicaremos mayor cosa sobre esto, solo que si instalamos dnsmasq **por supuesto será que será él el encargado de ello**. Sin embargo si quieren ver en acción a este software systemd-resolve, en una ventana de comandos ejecuten el siguiente comando, déjenlo abierto y naveguen, usen **dig** con algún dominio, lean correo con Thunderbird, hagan ping en otra terminal, etcétera:

```
journalctl -u systemd-resolved -f
```

También pueden ver el estado del servicio con:

```
systemd-resolve --status
```

Siéntanse a gusto de probar todo esto, porque el siguiente paso es inhabilitar a systemd-resolve para poder instalar a dnsmasq.

Desinstalando systemd-resolve

Con la primera línea inhabilitamos su arranque y con la segunda detenemos el servicio en sí:

```
sudo systemctl disable systemd-resolved sudo systemctl stop systemd-resolved
```

Luego vamos a seguir el enlace simbólico del fichero **resolv.conf** para saber hacia donde apunta y revisamos lo que contiene su contraparte:

...y como pueden ver en la figura anterior, todo se reduce a un reenvio a la famosa dirección 127.0.0.53

Lo siguiente que haremos será eliminar ese enlace simbólico, borrar el archivo **/etc/resolv.conf** y proceder a crear uno nuevo con el comando **echo** apuntando al DNS que ustedes prefieran (nosotros antes usábamos los DNS de Norton pero como el servicio fue eliminado pues ahora estamos usando los DNS de Google mientras encontramos un sustituto).

https://twitter.com/ks7000/status/789544726454206464

https://twitter.com/ks7000/status/880366908742217728

Hechas todas estas aclaratorias, procedemos entonces:

```
ls -lh /etc/resolv.conf sudo rm /etc/resolv.conf echo "nameserver 8.8.8.8" > /etc/resolv.conf echo "nameserver 8.8.4.4" >> /etc/resolv.conf
```

Instalando dnsmasq

Como dijimos al principio, si ya verificamos que los repositorios que tenemos configurados en nuestro sistema contienen la versión 2.78 o superior (**apt show dnsmasq**) simplemente:

```
sudo apt update sudo apt install dnsmasq
```

De ser necesario el caso, podremos descargar directamente el paquete .deb e instalarlo con las siguientes instrucciones:

```
wget http://archive.ubuntu.com/ubuntu/pool/universe/d/dnsmasq/dnsmasq_2.79-1_all.deb ls -la *.deb sudo dpkg -i dnsmasq_2.79-1_all.deb
```

Con la primera línea descargamos la versión 2.79 (siempre revisen si hay versión nueva), con la segunda nos aseguramos de que se haya descargado (son 18 kilobytes, bien pequeño) y con la tercera lo instalamos en sí.

¿Recuerdan el fichero **/etc/resolv.conf**, donde colocamos nuestros DNS preferidos? Pues bien, probablemente ya tengamos instalada la librería **openresolv** (versión 3.3.0-1 al escribir estas líneas) que permite ejecutar los *demonios* que permiten configurar y administrar dicho fichero. Si no la tenemos instalada de seguro que al instalar dnsmasq se incluye.

Por otra parte la librería que **siempre** se instala es **dnsmasq-base**, imaginen ustedes que es pequeño el programa y sus librerías, más pequeñas aun, son utilizadas por otros programas... La tercera librearía básica es **network-manager** una potente herramienta de los sistemas operativos GNU/Linux, básicamente se encarga de mantener bajo control las diferentes tarjetas de red (recordemos que en este siglo XXI aparte de ethernet existe Wi-fi, LTE, etcétera). Para no extendernos les colocamos su descripción exacta en idioma inglés:

> NetworkManager is a system network service that manages your network devices and connections, attempting to keep active network connectivity when available. It manages ethernet, WiFi, mobile broadband (WWAN), and PPPoE devices, and provides VPN integration with a variety of different VPN services.

Otras dependencias

- Si programamos con el lenguaje LUA podremos hacer guiones para administrar DHCP con esta herramienta (**sudo apt install dnsmasq-base-lua**).
- Volviendo al tema de DHCP, si lo usaremos con dnsmasq pues debemos incluir : **sudo apt install dnsmasq-utils**

Comprobando a dnsmasq

Su funcionamiento es sencillo de comprobar de dos maneras:

```
dig pandorafms.org @localhost nslookup wikipedia.org @localhost
```

Como vemos en ambas le estamos indicando al comando que utilicen al equipo local como "resolvedor" y no a los DNS comunes que tengamos configurados. De nuevo, según lo que tengamos en **/etc/resolv.conf** nos redireccionará; sabemos que es como tedioso o complicado (y hasta absurdo) pero así es la programación y la computación.

Ahora bien, la comprobación correcta que nos informará de forma concreta cómo se ejecuta dnsmasq sería...

Permitiendo al cortafuegos

Antes de continuar debemos permitir que el cortafuegos, en este caso **ufw** *(Ubuntu Fire Wall)* abra el puerto 53:

```
sudo ufw allow 22 sudo ufw allow 53
```

¿Pero de qué vamos, y ese puerto 22? Pues debemos asegurarnos, si nos conectamos remoto -lo más seguro- con **ssh**, de que ese puerto quede abierto para nosotros. Si utilizan cualquier otro número de puerto lo mejor sería **sudo ufw allow ssh** (esta última acción de manera análoga no lo podremos hacer con dnsmasq porque opera en otro nivel distinto).

Comprobando, de nuevo, a dnsmasq

La instrucción que nos permitirá conocer a fondo si funciona y cómo funciona dnsmasq, su estado, sería:

```
systemctl status dnsmasq.service
```

Y veríamos lago mu parecido a esto:

Configurando dnsmasq

Como cosa rar en GNU/Linux (sarcasmo) todo está un un fichero: **/etc/dnsmasq.conf** así que ya sabemos cómo editar un archivo de texto cualquiera, *colocaremos ahora las secciones más relevantes y deberán usar la opción de búsqueda de palabras clave de dicho editor de texto.*

Escogiendo puerto de escucha

Que ya dijimos que es el 53, pero si queremos usar uno distinto, por ejemplo el 5353 que no está en la lista de puertos comunes, pues colocamos:

```
port=5353
```

O el puerto deseado. Si **no vamos a usar a dnsmasq como DNS** (sí, eso se escucha ilógico y hasta contraproducente) podemos colocar:

```
port=0
```

Por si lo preguntan, el escenario anterior se daría únicamente si dnsmasq solo cumple funciones de DHCP y/o TFTP.

Filtrando errores de tipeo

Sí, que muchos usuarios fueron mal acostumbrados por los navegadores web a realizar búsquedas desde la barra de direcciones e incluso en Firefox ocultan el cuadro de texto para las búsquedas normales, comunes y corrientes:

Por eso debemos filtrar la cadena de texto que envía el navegador web (supuesto dominio solicitado) y verificar que tiene puntos y que cumple con las normas de nombrado de un dominio web.

```
domain-needed
```

Solamente escriban eso (quitando el "#" al principio de la línea) para no fatigar a otros DNS con solicitudes basura. Noten que no necesita el signo de igualdad y mucho menos un valor específico.

Escogiendo la tarjeta de red

Fuentes Consultadas

En idioma inglés

- «How to Install and Configure Dnsmasq on Ubuntu 18.04 LTS» by Josphat Mutai (WayBack Machine).
- «SYSTEMD-RESOLVED.SERVICE(8)systemd-resolved.serviceYSTEMD-RESOLVED.SERVICE(8)» (WayBack Machine).
- «HowTo dnsmasq» (WayBack Machine).

Vulnerabilidades

- «Pwning dnsmasq (part 2) with ROP» by Alejandro Cáceres (WayBack Machine).
- «dnsmasq » (WayBack Machine).
- «Recent Dnsmasq Vulnerabilities Explained» by Alex Zelivansky (WayBack Machine).
- «How to protect your systems from newly-discovered Dnsmasq vulnerabilities» by Scott Matteson (WayBack Machine).
- «Dnsmasq: A Reality Check and Remediation Practices» by Federico Maggi (WayBack Machine).
- «Using dnsmasq systemd-resolved resolvconf» (WayBack Machine).

Repositorios

- «https://ubuntu.pkgs.org/18.04/ubuntu-universe-i386/dnsmasq_2.79-1_all.deb.html»

En idioma japonés

- «OpenShift???DNS??????...??????dnsmasq?gdb????»