I always want to get rid of the sh*tload of ads/spywares/analitics/<your name it> crap downloaded by your browser.
I have a solution on the DNS with Bind (using only one url to get an hosts file) see DNS on Raspberry Pi and DNS on OpenBSD.
I now use 2 DNS at home. One running Bind with default config. Another running dnsmasq for my home dns entry and the ease to add all those funky domain to hell.

I use 2 scripts, one to generate a file containing all domains to blacklist. And a second one to merge it with my DNS entries + whitelist some of them.
Structure is as follow:

- gen_domain_list.sh
- gen_hosts-dnsmasq.sh
- dl/ (folder containing the downloaded host files - erased before each run)
- list/
  - dns.list
  - white.list

```
#! /bin/sh
# Script that generate a list of domain to blacklist

# First rm previous version of downloaded and generated files
echo ">> Cleaning of old files"
rm dl/*

# Get the hosts files
echo ">> Getting all lists"
wget -O dl/HOST1.txt http://winhelp2002.mvps.org/hosts.txt
wget -O dl/HOST2.txt
http://sourceforge.net/projects/adzhosts/files/FORADAWAY.txt/download
wget -O dl/HOST3.txt http://www.securemecca.com/Downloads/hosts.txt
wget -O dl/HOST4.txt http://someonewhocares.org/hosts/hosts
wget -O dl/HOST5.txt http://adaway.org/hosts.txt
wget -O dl/HOST6.txt http://hosts-file.net/ad_servers.asp
wget -O dl/HOST7.txt
"http://pgl.yoyo.org/adservers/serverlist.php?hostformat=&showintro=1&startd
ate%5Bday%5D=&startdate%5Bmonth%5D=&startdate%5Byear%5D=&mimetype=plaintext"
wget -O dl/HOST8.txt http://sysctl.org/cameleon/hosts
wget -O dl/HOST9.txt http://www.malwaredomainlist.com/hostslist/hosts.txt
wget -O dl/HOST10.txt http://www.hostsfile.org/Downloads/hosts.txt
wget -O dl/HOST11.txt "http://adblock.gjtech.net/?format=hostfile"


# Process files
echo ">> Generating a unique hosts file"
for host_file in `ls ./dl/*.txt`
  do
    echo "Processing $host_file"
    echo "Converting to Unix format…"
    dos2unix $host_file
```

```
    echo "Removing obsolete lines"
    # Removes comments and empty lines
    sed -i 's/#.*$//' $host_file
    sed -i '/^\s*$/d' $host_file
    # Replaces tabulation by space
    sed -i "s/\t/ /ig" $host_file
    # Removing lines containing ::1 localhost broadcasthost
    sed -i '/::1/d' $host_file
    sed -i '/localhost/d' $host_file
    sed -i '/broadcasthost/d' $host_file
    # Removing 0.0.0.0
    sed -i 's/0\.0\.0\.0 //g' $host_file
    # Removing 127.0.0.1 as well as mistakes like 127.0.1.1
    sed -i 's/127\.0\..\.1 //g' $host_file
    # Removing trailing space
    sed -i 's/ *$//' $host_file
    # Send the result to a temp file
    cat $host_file >> dl/HOSTS.tmp
done

echo ">> Arranging the file"
# remove the www. in front of domains
#sed -i 's/www\.//' dl/HOSTS.tmp
# Lowercase everything
tr '[:upper:]' '[:lower:]' < dl/HOSTS.tmp > dl/HOSTS.low
# sort unique domains
sort -u -o hosts.sec dl/HOSTS.low
echo ">> File hosts.sec generated"
```

```
#!/bin/sh
# Script that generate a hosts file for dnsmasq (first DNS entries then
blacklisted domains)
# requires dns.list dns list
# requires white.list domain white listed

rm hosts hosts.add

# Concatenating files
echo "Adding dns.list"
cat list/dns.list > hosts
echo "Whitelisting hosts.sec with white.list"
comm -23 hosts.sec list/white.list >> hosts.add
# Add 0.0.0.0 to all the lines
sed -i -e 's/^/0\.0\.0\.0 /' hosts.add
# Add all those #%$@ domains to the hosts file
cat hosts.add >> hosts
echo "Done!"
```

Content of dns.list

```
127.0.0.1       localhost
```

```
::1             localhost ip6-localhost ip6-loopback
fe00::0         ip6-localnet
ff00::0         ip6-mcastprefix
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters


#network devices and Services [1-30]
192.168.1.1     router
192.168.1.2     dns
192.168.1.3     wifi
```

Content of white.list

```
24heures.ch
anibis.ch
beagleboard.org
billionuploads.com
bluewin.ch
fdj.fr
```

And voilà :)

From:
https://wiki.fortier-family.com/ - **Warnaud's Wiki**

Permanent link:
https://wiki.fortier-family.com/scripting/bash/hosts

Last update: **2021/12/29 21:03**