



EasyList Tracker and Adblocks to Proxy Auto Configuration (PAC) File and Privoxy Actions and Filters

#proxy-configuration #pac #tracker #privacy-enhancing-technologies #privacy-tools #easylist #privoxy #pac-files #proxy #adblock #adblocking #lan

69 commits 1 branch 0 releases 1 contributor GPL-3.0

Branch: master New pull request Find file

Clone or download

File	Commit	Date
adblock2privoxy @ 5b7c073	Update adblock2privoxy	Apr 16, 2018
.gitignore	Initial commit	Jun 22, 2017
.gitmodules	Add adblock2privoxy repo as submodule	Jul 19, 2017
LICENSE	Initial commit	Jun 22, 2017
README.md	Update README.md	Apr 16, 2018
easylist_pac.py	Anti-Facebook rules	Apr 4, 2018
proxy.pac	Anti-Facebook rules	Apr 4, 2018

README.md

easylist-pac-privacy

EasyList Tracker and Adblock Rules to Proxy Auto Configuration (PAC) File and Privoxy Actions and Filters

Converts EasyList tracker and ad blocking rules to efficient network-level blocks in a proxy.pac file for automatic proxy network configurations and Privoxy proxy servers.

Easily incorporates multiple blocking rulesets into both PAC and Privoxy formats, including easyprivacy.txt, easylist.txt, fanboy-annoyance.txt, fanboy-social.txt, antiadblockfilters.txt, malwaredomains_full.txt, and the anti-spamware list adblock-list.txt.

Purpose

Provide tracker and ad blocking at the kernel and network layers using the crowd-sourced EasyList blocking rulesets used by client-based browser plugins. This proxy configuration provides EasyList blocking rules for all devices on the LAN or VPN, beyond the capabilities of client-specific plugins.

A combination of a proxy.pac file with Privoxy and a webserver for CSS rules that perform element blocking is used to implement all the features of EasyList blocking rules.

Blocking capability	Browser Plugin	proxy.pac	Privoxy	Privoxy+CSS
EasyList regex rules	✓	✓	✓	✓
EasyList element hiding	✓	✗	✗	✓
HTTP	✓	✓	✓	✓
HTTPS	✓	✓	✗	✗
Client-level	✓	✓	✓	✓
Kernel-level	✗	✓	✓	✓
Network-level	✗	✓	✓	✓
Large rulesets	✓	✗	✓	✓

Proxy Auto Configuration (PAC)

To Use: Localhost

Download the proxy.pac file.

On macOS (without Server.app):

```
sudo cp ~/Downloads/proxy.pac /Library/WebServer/Documents
sudo apachectl start
```

Set your network Proxy Auto Configuration setting to:

```
http://localhost/proxy.pac or http://host-ip-address/proxy.pac
```

Advantages

- Works for any mobile or desktop device on your LAN.
- Works with an upstream proxy if specified in the proxy.pac file.
- Individual update control and customization of the proxy.pac file and filter rules.
- Possible internet access if port 80 exposed outside the LAN firewall.

Disadvantages

- Does not work on mobile data networks.
- No internet access unless port forwarding to host is used.

To Use: VPN

Configure an OpenVPN to use the proxy.pac file hosted on your LAN.

This is the best option.

Advantages

- Works on any mobile or desktop device on any mobile data or WiFi network worldwide.
- Individual update control and customization of the proxy.pac file and filter rules.
- Security and privacy benefits of VPNs.

Disadvantages

- Necessity of VPN server.

To Use: GitHub Host

Set your network Proxy Auto Configuration setting to:

```
https://raw.githubusercontent.com/essandess/easylist-pac-privacy/master/proxy.pac
```

Advantages

- Works on any mobile or desktop device on any WiFi network worldwide.
- GitHub server; private web server not necessary.

Disadvantages

- Does not work on mobile data networks.
- Does not work on iOS without an open blackhole with HTTP return code 200 for blackholed sites.
- Reliance on a third-party (me) for pass/block rule sets, updates, and proxy.pac integrity.

Details

Using EasyList rules in a proxy.pac file provides these benefits:

- Tracker and Ad blocking performed in all clients that use PAC files, browsers and non-browsers alike.
- Tracker and Ad blocking on both desktop and mobile devices, especially via VPN.
- Browser plugins or filtering proxies are not necessarily used (although PAC files work well in sequence with these).
- PAC files do not alter the webpage DOM, used by adblock detection methods.

The script easylist_pac.py downloads EasyList and EasyPrivacy rules and converts these to a combination of very efficient Javascript hash lookups and efficient NFA regular expressions. The size of the PAC file and rulesets are limited in the posted example to a total of over fifteen thousand (18788) to ensure efficient execution on modern mobile devices. For full rulesets, use in conjunction with a browser plugin and/or Privoxy.

Example hash (exact match) blocking entries look like:

```
"tracker.myseofriend.net"
"adwitracker.fwix.com"
```

Example regular expression blocking rules look like:

```
online.*/promoredirect?key=
secureprovide1.com/*=tracking
```

EasyList to proxy.pac converter

```
python3 easylist_pac.py
python3 easylist_pac.py -h
python3 easylist_pac.py -b blackhole-ip-address:port -d download_dir -p proxy:port -P proxy.pac.orig
```

The new file proxy.pac will be created in the (default ~/Downloads directory. See easylist_pac.py -h for options.

Privoxy

The repo adblock2privoxy is used to achieve nearly full EasyList rule capability, complete with element hiding.

After installing adblock2privoxy, an example production run with regular updates looks like:

```
adblock2privoxy -p /usr/local/etc/adblock2privoxy/privoxy -w /usr/local/etc/adblock2privoxy/css -d
10.0.1.3:8119 \
  https://easylist.to/easylist/easyprivacy.txt \
  https://easylist.to/easylist/easylist.txt \
  https://easylist.to/easylist/fanboy-annoyance.txt \
  https://easylist.to/easylist/fanboy-social.txt \
  https://easylist-downloads.adblockplus.org/antiadblockfilters.txt \
  https://easylist-downloads.adblockplus.org/malwaredomains_full.txt \
  https://raw.githubusercontent.com/Dawsey21/Lists/master/adblock-list.txt

# then every few days
adblock2privoxy -t /usr/local/etc/adblock2privoxy/privoxy/ab2p.task
# restart privoxy, e.g. sudo port unload privoxy ; sudo port load privoxy
```

Public Service Announcement

This proxy.pac is configured to block all known tracker and adware content at the network level. Many websites now offer an additional way to block ads: subscribe to their content. Security and privacy will always necessitate ad blocking, but now that this software has become mainstream with mainstream effects, ad blocker users must consider the potential impact of ad blocking on the writers and publications that are important to them. Personally, two publications that I gladly pay for, especially for their important US political and other coverage, are the New York Times and The Atlantic. I encourage all users to subscribe to their own preferred publications and writers.