```
docker pull jacklul/pihole
```

Update Pi-hole's lists from remote sources

When using remote lists like this or this it's a hassle to manually check for changes and update - this script will do that for you!

User-created entries will not be touched and those removed from the remote list will be disabled instead.

If you're not using remote lists like the ones mentioned above then this script will be useless to you - Pi-hole already updates the lists weekly automatically.

Requirements

- **Pi-hole v5+** installed (fresh install preferred)

- **php-cli >=7.0** and **a few extensions** (`sudo apt-get install php-cli php-sqlite3 php-intl php-curl`)

- **systemd** is optional but recommended

Install

Docker users - look below.

This command will install this script to `/usr/local/sbin` :

```
wget -O - https://raw.githubusercontent.com/jacklul/pihole-updatelists/master/install.sh | sudo bash
```

Alternatively you can clone this repo and `sudo bash install.sh` .

If **systemd** is available this will also add service and timer unit files to the system, otherwise a crontab entry in `/etc/cron.d/pihole-updatelists` will be created.

In the future to quickly update the script you can use `sudo pihole-updatelists --update` .

Note that in most cases you will be able to execute this script globally as `pihole-updatelists` command but some will require you to add `/usr/local/sbin` to `$PATH` or execute it via `/usr/local/sbin/pihole-updatelists` .

This script does nothing by default (except running `pihole updateGravity`), you have to configure it.

Disable default gravity update schedule

If you don't plan on updating adlists or want to keep Pi-hole's gravity update schedule you should skip this section and set `UPDATE_GRAVITY=false` in the configuration file.

You should disable entry with `pihole updateGravity` command in `/etc/cron.d/pihole` as this script already runs it:

```
sudo nano /etc/cron.d/pihole
```

Put a `#` before this line (numbers might be different):

```
#49 4   * * 7   root    PATH="$PATH:/usr/local/bin/" pihole updateGravity >/var/log/pihole_updateGravity.log || cat /var/log/pihole_updateGravity.log
```

Alternatively, the following `sed` command will disable the same entry:

```
sudo sed -e '/pihole updateGravity/ s/^#*/#/' -i /etc/cron.d/pihole
```

You might have to do this after each Pi-hole update.

Migrating lists and domains

If you already imported any of the remote lists manually you should migrate their entries to allow the script to disable them in case they are removed from the remote list.

If you used pihole5-list-tool to import adlists and whitelist you can use these commands to do this quickly:

```
sudo sqlite3 /etc/pihole/gravity.db "UPDATE adlist SET comment = 'Managed by pihole-updatelists' WHERE comment LIKE '%Firebog |%' OR comment LIKE '%[ph5lt]'"
sudo sqlite3 /etc/pihole/gravity.db "UPDATE domainlist SET comment = 'Managed by pihole-updatelists' WHERE comment LIKE '%AndeepND |%' OR comment LIKE '%[ph5lt]'"
```

(code up to date as of pihole5-list-tool 0.6.0)

Alternatively, some manual work is required - pick one:

- Manually modify comment field of all imported domains/adlists to match the one this script uses (see `COMMENT` variable in **Configuration** section) **(recommended but might be a lot of work)**

- Manually delete all imported domains/adlists from the web interface (might be a lot of work)

- Wipe all adlists and domains (not recommended but fast - use this if you want to start fresh)

    - backup your lists and custom entries (write them down somewhere, do not use the Teleporter)

    - run the following commands:

```
            sudo sqlite3 /etc/pihole/gravity.db "DELETE FROM adlist"
            sudo sqlite3 /etc/pihole/gravity.db "DELETE FROM adlist_by_group"
            sudo sqlite3 /etc/pihole/gravity.db "DELETE FROM domainlist"
            sudo sqlite3 /etc/pihole/gravity.db "DELETE FROM domainlist_by_group"
```

- keep reading and configure the script then run `sudo pihole-updatelists` to finish up

- (only when `UPDATE_GRAVITY=false`) run `pihole updateGravity`

Install with Docker

Follow the official instructions and add a volume for `/etc/pihole-updatelists/` directory.

If you need to pull a specific version of Pi-hole image you have no other choice but to use custom Dockerfile.

Using custom image

Use `jacklul/pihole:latest` image instead of `pihole/pihole:latest` .

Using official image

If you don't want to use my image you can write custom `Dockerfile` :

```
FROM pihole/pihole:latest

RUN apt-get update && apt-get install -Vy php-cli php-sqlite3 php-intl php-curl

RUN wget -O - https://raw.githubusercontent.com/jacklul/pihole-updatelists/master/install.sh | bash
```

Then build your image locally and use that image in your `docker-composer.yml` or launch command line. You will have to update your local image manually each time update is released.

Container Configuration

Your `docker-compose.yml` file should look similar to this:

```
version: "3"

services:
  pihole:
    container_name: pihole
    image: jacklul/pihole:latest
    ports:
      - "53:53/tcp"
      - "53:53/udp"
      - "67:67/udp"
      - "80:80/tcp"
      - "443:443/tcp"
    environment:
      TZ: 'America/Chicago'
    volumes:
      - './etc-pihole/:/etc/pihole/'
      - './etc-dnsmasq.d/:/etc/dnsmasq.d/'
      - './etc-pihole-updatelists/:/etc/pihole-updatelists/'
    dns:
      - 127.0.0.1
      - 1.1.1.1
    cap_add:
      - NET_ADMIN
    restart: unless-stopped
```

If you already have existing `gravity.db` you should also check out Migrating lists and domains section, keep in mind that you will have to adjust paths in the commands mentioned there.

Configuration

Default configuration file is `/etc/pihole-updatelists.conf` .

```
sudo nano /etc/pihole-updatelists.conf
```

Available variables

Variable	Default	Description
ADLISTS_URL	" "	Remote list URL containing list of adlists to import
WHITELIST_URL	" "	Remote list URL containing exact domains to whitelist
REGEX_WHITELIST_URL	" "	Remote list URL containing regex rules for whitelisting
BLACKLIST_URL	" "	Remote list URL containing exact domains to blacklist This is specifically for handcrafted lists only, do not use regular blocklists here!
REGEX_BLACKLIST_URL	" "	Remote list URL containing regex rules for blacklisting
COMMENT	"Managed by pihole-updatelists"	Comment string used to know which entries were created by the script You can still add your own comments to individual entries as long you keep this string intact

Variable	Default	Description
GROUP_ID	0	Assign additional group to all inserted entries, to assign only the specified group (do not add to the default) make the number negative `0` is the default group, you can view ID of the group in Pi-hole's web interface by hovering mouse cursor over group name field on the 'Group management' page
PERSISTENT_GROUP	false	Makes sure entries have the specified group assigned on each run This does not prevent you from assigning more groups through the web interface but can remove entries from the default group if GROUP_ID is negative number Do not enable this when you're running multiple different configs **This option will be removed in the future version!**
REQUIRE_COMMENT	true	Prevent touching entries not created by this script by comparing comment field When `false` any user-created entry will be disabled
UPDATE_GRAVITY	true	Update gravity after lists are updated? (runs `pihole updateGravity`) When `false` invokes lists reload instead Set to `null` to do nothing
VACUUM_DATABASE	false	Vacuum database at the end? (runs `VACUUM` SQLite command) Will cause additional writes to disk **This option will be removed in the future version!**
VERBOSE	false	Show more information while the script is running
DEBUG	false	Show debug messages for troubleshooting purposes
DOWNLOAD_TIMEOUT	60	Maximum time in seconds one list download can take before giving up You should increase this when downloads fail because of timeout
IGNORE_DOWNLOAD_FAILURE	false	Ignore download failures when using multiple lists This will cause entries from the lists that failed to download to be disabled
GRAVITY_DB	"/etc/pihole/gravity.db"	Path to `gravity.db` in case you need to change it
LOCK_FILE	"/var/lock/pihole-updatelists.lock"	Process lockfile to prevent multiple instances of the script from running You shouldn't change it - unless `/var/lock` is unavailable
LOG_FILE	" "	Log console output to file In most cases you don't have to set this as you can view full log in the system journal Put `-` before path to overwrite file instead of appending to it
GIT_BRANCH	"master"	Branch to pull remote checksum and update from

String values should be put between `" "` , otherwise weird things might happen.

You can also give paths to the local files instead of URLs, for example setting `WHITELIST_URL` to `/home/pi/whitelist.txt` will fetch this file from filesystem.

Multiple configurations

You can specify alternative config file by passing the path to the script through `config` parameter: `pihole-updatelists --config=/home/pi/pihole-updatelists2.conf` - this combined with different `COMMENT` string can allow multiple script configurations for the same Pi-hole instance.

Alternatively, you could try config_sections branch which currently has test version that can achieve the same configuration in just one config file.

Multiple list URLs

You can pass multiple URLs to the list variables by separating them with whitespace (space or new line):

```
ADLISTS_URL="https://v.firebog.net/hosts/lists.php?type=tick  https://raw.githubusercontent.com/you/adlists/master/my_adlists.txt"
```

If one of the lists fails to download nothing will be affected for that list type.

Recommended lists

List	URL/Variable value	Description
Adlist (ADLISTS_URL)	https://v.firebog.net/hosts/lists.php?type=tick	https://firebog.net - safe lists only
Whitelist (WHITELIST_URL)	https://raw.githubusercontent.com/anudeepND/whitelist/master/domains/whitelist.txt	https://github.com/anudeepND/whitelist - commonly whitelisted
Regex blacklist (REGEX_BLACKLIST_URL)	https://raw.githubusercontent.com/mmotti/pihole-regex/master/regex.list	https://github.com/mmotti/pihole-regex - basic regex rules

Please note that mmotti/pihole-regex list can sometimes block domains that should not be blocked - any false positives should be reported to the repository to be included in the whitelist (in that case you might consider adding that list to the `WHITELIST_URL` too).

Extra information

Runtime options

These can be used when executing `pihole-updatelists` .

Option	Description
`--help, -h`	Show help message, which is simply this list
`--no-gravity, -n`	Force gravity update to be skipped
`--no-reload, -b`	Force lists reload to be skipped Only if gravity update is disabled either by configuration (`UPDATE_GRAVITY=false`) or `--no-gravity` parameter

Option	Description
`--no-vacuum, -m`	Force database vacuuming to be skipped **This parameter will be removed in the future version!**
`--verbose, -v`	Turn on verbose mode
`--debug, -d`	Turn on debug mode
`--config=<file>`	Load alternative configuration file
`--git-branch=<branch>`	Select git branch to pull remote checksum and update from Can only be used with `--update` and `--version`
`--update`	Update the script using selected git branch
`--force`	Force update without checking for newest version
`--version`	Show script checksum (and also if update is available)

Changing the schedule

By default, the script runs at random time (between 03:00 and 04:00) on Saturday, to change it you'll have to override timer unit file:

```
sudo systemctl edit pihole-updatelists.timer
```

```
[Timer]
RandomizedDelaySec=5m
OnCalendar=
OnCalendar=Sat *-*-* 00:00:00
```

If systemd is not available you just modify the crontab entry in `/etc/cron.d/pihole-updatelists` :

```
14 6 * * 6   root   /usr/local/sbin/pihole-updatelists
```

Running custom commands before/after scheduled run

Override service unit file:

```
sudo systemctl edit pihole-updatelists.service
```

```
[Service]
Type=oneshot
ExecStartPre=echo "before"
ExecStartPost=echo "after"
```

If systemd is not available you just modify the crontab entry in `/etc/cron.d/pihole-updatelists` :

```
30 3 * * 6   root   /home/pi/before.sh && /usr/local/sbin/pihole-updatelists && /home/pi/after.sh
```

You can use `;` instead of `&&` if you don't want the execution to stop on previous command failure.

Changing comment value after running the script

```
sudo sqlite3 /etc/pihole/gravity.db "UPDATE adlist SET comment = 'NEWCOMMENT' WHERE comment LIKE '%Managed by pihole-updatelists%'"
sudo sqlite3 /etc/pihole/gravity.db "UPDATE domainlist SET comment = 'NEWCOMMENT' WHERE comment LIKE '%Managed by pihole-updatelists%'"
```

Replace `NEWCOMMENT` with your new desired comment value. This assumes `Managed by pihole-updatelists` is the old comment value, replace it with your old custom value when needed.

Custom comments for entries

If you wish to add custom comments to entries you can use the following file syntax:

```
example-domain.com # your comment
```

Which will cause `example-domain.com` to have `comment` set to `your comment | Managed by pihole-updatelists` .

You can also add your comments directly through the Pi-hole's web interface by either appending or prepending the comment field for entries.

Uninstalling

```
wget -O - https://raw.githubusercontent.com/jacklul/pihole-updatelists/master/install.sh | sudo bash /dev/stdin uninstall
```

or remove files manually:

```
sudo rm -vf /usr/local/sbin/pihole-updatelists /etc/bash_completion.d/pihole-updatelists /etc/systemd/system/pihole-updatelists.service /etc/systemd/system/pihole-updatelists.timer /etc/cron.d/piho
```

License

MIT License.

jacklul/pihole ★

By **jacklul** • Updated 4 days ago

Pi-hole image combined with pihole-updatelists script

Container

⬇ Pulls **500K+**

Overview   **Tags**

Sort by Newest ▾   🔍 Filter Tags

TAG

latest

Last pushed **4 days ago** by **jacklul**

docker pull jacklul/pihole:latest   ⧉

DIGEST	OS/ARCH	COMPRESSED SIZE ⓘ
0ca6f8018ae0	linux/386	178.55 MB
34ff108ef0da	linux/amd64	145.46 MB
7a2efdd140e3	linux/arm/v7	165.96 MB
b01c7654a52a	linux/arm64	176.27 MB

TAG

dev

Last pushed **4 days ago** by **jacklul**

docker pull jacklul/pihole:dev   ⧉

DIGEST	OS/ARCH	COMPRESSED SIZE ⓘ
201fa6b8d410	linux/386	178.55 MB
7e89117ce678	linux/amd64	145.46 MB
84e6d6ac5c31	linux/arm/v7	165.96 MB
c11ae7ba9018	linux/arm64	176.27 MB