

Ubuntu (Server) PiHole Installation

<https://docs.pi-hole.net/guides/dns-over-https/>

PiHole Installation

```
wget -O basic-install.sh https://install.pi-hole.net
sudo bash basic-install.sh
```

Nach der Installation

Change admin password using terminal.

```
$ pihole -a -p
Enter New Password (Blank for no password): *****
Confirm Password: *****
New password set
```

Cloudflared Installation

```
wget https://bin.equinox.io/c/VdrWdbjqyF/cloudflared-stable-linux-amd64.deb
sudo apt-get install ./cloudflared-stable-linux-amd64.deb
cloudflared -v
```

Create a cloudflared user to run the daemon.

```
sudo useradd -s /usr/sbin/nologin -r -M cloudflared
```

proceed to create a configuration file for cloudflared by copying the following in to /etc/default/cloudflared. This file contains the command-line options that get passed to cloudflared on startup.

```
Sudo vi /etc/default/cloudflared
```

INSERT:

```
# Commandline args for cloudflared
CLOUDFLARED_OPTS=--port 5053 --upstream https://1.1.1.1/dns-query --upstream
https://1.0.0.1/dns-query
```

Update the permissions for the configuration file and cloudflared binary to allow access for the cloudflared user

```
sudo chown cloudflared:cloudflared /etc/default/cloudflared
sudo chown cloudflared:cloudflared /usr/local/bin/cloudflared
```

Then create the systemd script by copying the following in to /lib/systemd/system/cloudflared.service. This will control the running of the service and allow it to run on startup.

```
sudo vi /lib/systemd/system/cloudflared.service
```

INSERT:

```
[Unit]
Description=cloudflared DNS over HTTPS proxy
After=syslog.target network-online.target

[Service]
Type=simple
User=cloudflared
EnvironmentFile=/etc/default/cloudflared
ExecStart=/usr/local/bin/cloudflared proxy-dns $CLOUDFLARED_OPTS
Restart=on-failure
RestartSec=10
KillMode=process

[Install]
WantedBy=multi-user.target
```

Enable the systemd service to run on startup, then start the service and check its status.

```
sudo systemctl enable cloudflared
sudo systemctl start cloudflared
sudo systemctl status cloudflared
```

Now test that it is working! Run the following dig command, a response should be returned similar to the one below

```
dig @127.0.0.1 -p 5053 google.com
```

```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> @127.0.0.1 -p 5053 google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65181
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1536
;; QUESTION SECTION:
;google.com.                IN A

;; ANSWER SECTION:
google.com.                299 IN A    243.65.127.221

;; Query time: 3 msec
;; SERVER: 127.0.0.1#5053(127.0.0.1)
;; MSG SIZE rcvd: 65
```

DNS Einstellung in der Admin WebGUI der PiHole:

Upstream DNS Servers

IPv4		IPv6		Name
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Google
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	OpenDNS
<input type="checkbox"/>	<input type="checkbox"/>			Level3
<input type="checkbox"/>	<input type="checkbox"/>			Norton
<input type="checkbox"/>	<input type="checkbox"/>			Comodo
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	DNS.WATCH
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Quad9
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cloudflare

Custom 1 (IPv4)

Custom 2 (IPv4)

Custom 3 (IPv6)

Custom 4 (IPv6)