

[hadisfr / shadowsocks\\_on\\_ubuntu.md](#)

Created 6 years ago • Report abuse

[Code](#) [Revisions](#) 1 [Stars](#) 2

a simple guide to run a shadowsocks proxy on Ubuntu

[shadowsocks\\_on\\_ubuntu.md](#)

# Run a Shadowsocks Proxy Server on Ubuntu

## Install Shadowsocks

It's possible to install `ss-server` by `apt` :

```
apt-get install shadowsocks-libev
```

by `pip` :

```
pip install shadowsocks
```

or build it from source:

```
sudo apt update
sudo apt install gettext build-essential autoconf libtool libpcrc3-dev asciidoc
git clone https://github.com/shadowsocks/shadowsocks-libev.git
cd shadowsocks-libev
git submodule update --init --recursive
./autogen.sh
./configure
make
make install
```

## Configure Server

First, we create a new system user for shadowsocks:

```
adduser --system --no-create-home --group shadowsocks
```

Then we create a folder for shadowsocks configurations, and add configuration file to it.

```
mkdir -m 755 /etc/shadowsocks  
touch /etc/shadowsocks/shadowsocks.json
```

Then we add contents to shadowsocks.json we just created.

```
{  
  "server": "192.0.0.1",  
  "server_port": 8388,  
  "password": "mypassword",  
  "timeout": 300,  
  "method": "aes-256-gcm",  
  "fast_open": true  
}
```

The details about this json file can be found [here in shadowsocks' wiki](#) and [here](#). As mentioned [here](#), `local_address` and `local_prot` should not be appeared in server-side config file.

## Optimize Kernel

You may want to apply some changes to `/etc/sysctl.d/local.conf` and run `sysctl -system` too. See [here](#).

## Open Ports on Firewall

We should open TCP connection to port specified in config file, 8388 in this example.

### iptables

```
iptables -4 -A INPUT -p tcp --dport 8388 -m comment --comment "Shadowsocks se
```

### ufw

```
ufw allow proto tcp to 0.0.0.0/0 port 8388 comment "Shadowsocks server listen
```

## Run Server

You should be able to run ss-server like this:

```
ss-server -c /etc/shadowsocks/shadowsocks.json -a shadowsocks -v start
```

## Create System Service

To make system daemon, we should create

/etc/systemd/system/shadowsocks.service with these contents:

```
[Unit]
Description=Shadowsocks proxy server

[Service]
User=root
Group=root
Type=simple
ExecStart=/usr/local/bin/ss-server -c /etc/shadowsocks/shadowsocks.json -a shadowsocks start
ExecStop=/usr/local/bin/ss-server -c /etc/shadowsocks/shadowsocks.json -a shadowsocks -v stop

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

And run:

```
systemctl daemon-reload
systemctl start shadowsocks
```

And if everything is OK, to make OS run service daemon automatically:

```
systemctl enable shadowsocks
```

## Run Client

To connect to server, download the appropriate client app from [shadowsocks website](#) and configure it based on shadowsocks.json you have created.

## Terminal Client

To run client on terminal, after installing `shadowsocks` or `shadowsocks-libev`, you should be able to run `sslocal` with an appropriate config file like this, as mentioned [here](#):

```
sslocal -c shadowsocks.json
```

Or:

```
sslocal -c shadowsocks.json -d start
```

Config file `shadowsocks.json` should be like this:

```
{
  "server": "my_server_ip",
  "server_port": 8388,
  "local_address": "127.0.0.1",
  "local_port": 1080,
  "password": "mypassword",
  "timeout": 300,
  "method": "aes-256-cfb",
  "fast_open": false
}
```

After that, you can connect to `socks5://127.0.0.1:1080` or tunnel all system's traffic throw it.

In terminal, can use:

```
export http_proxy=socks5://127.0.0.1:1080
export https_proxy=socks5://127.0.0.1:1080
```