Expose Documentation

- Introduction
- Upgrade from Expose v1

Getting Started
- Installation
- Share your first site
- Setup Your Token
- Questions & Issues

Client
- Sharing your local sites
- Local Dashboard
- Sharing TCP ports
- Basic Authentication
- Global Server Infrastructure

Server
- Starting the server
- SSL Support
- Admin Interface
- Server Configuration

Extending The Server
- Subdomain Generator
- User Repository

Starting the server

#

Expose is open source and you can host your own Expose server in your own infrastructure. The open source core contains the server and the client but does not include the Expose platform where you can manage your team, reserve subdomains or add custom domains via the web interface.

The expose binary that you install via composer contains both the server and the client, so you do not need any additional software for this to work.

Once you have successfully downloaded expose, you can start the server using this command:

```
expose serve my-domain.com
```

This will start listening for incoming expose client connections on port 8080 by default.

If you want, you can customize the port:

```
expose serve my-domain.com --port=3000
```

Validating auth tokens

#

When you start your expose server, anyone is able to connect to it by default. If you want to restrict your server only to users that have a valid "authentication token", you can start the server with the `--validateAuthTokens` option:

```
expose serve my-domain.com --validateAuthTokens
```

Don't worry - you can also change this later on through the admin interface.

Keeping the expose server running with supervisord

#

The `expose serve` daemon needs to always be running in order to accept connections. This is a prime use case for `supervisor`, a task runner on Linux.

First, make sure `supervisor` is installed.

```
# On Debian / Ubuntu
apt install supervisor

# On Red Hat / CentOS
yum install supervisor
systemctl enable supervisord

# On Mac
brew install supervisor
```

Once installed, add a new process that `supervisor` needs to keep running. You place your configurations in the `/etc/supervisor/conf.d` (Debian/Ubuntu) or `/etc/supervisord.d` (Red Hat/CentOS) directory.

Within that directory, create a new file called `expose.conf`.

```
[program:expose]
command=/usr/bin/php /home/expose/expose serve
numprocs=1
autostart=true
autorestart=true
user=forge
```

Once created, instruct `supervisor` to reload its configuration files (without impacting the already running `supervisor` jobs).

```
supervisorctl update
supervisorctl start expose
```

Your expose server should now be running (you can verify this with `supervisorctl status`). If it were to crash, `supervisor` will automatically restart it.

Please note that, by default, `supervisor` will force a maximum number of open files onto all the processes that it manages. This is configured by the `minfds` parameter in `supervisord.conf`.

If you want to increase the maximum number of open files, you may do so in `/etc/supervisor/supervisord.conf` (Debian/Ubuntu) or `/etc/supervisord.conf` (Red Hat/CentOS):

```
[supervisord]
minfds=10240; (min. avail startup file descriptors;default 1024)
```

After changing this setting, you'll need to restart the supervisor process (which in turn will restart all your processes that it manages).

Connecting the client

#

To configure a client to connect to your custom server, first publish the configuration file on the client. Once that is done, you can change the `host` and `port` configuration values on your client.

```
return [

   /*
   |--------------------------------------------------------------------------
   | Host
   |--------------------------------------------------------------------------
   |
   | The expose server to connect to. By default, expose is using the free
   | sharedwithexpose.com server, offered by Beyond Code. You will need a free
   | Beyond Code account in order to authenticate with the server.
   | Feel free to host your own server and change this value.
   |
   */
   'host' => 'my-domain.com',

   /*
   |--------------------------------------------------------------------------
   | Port
   |--------------------------------------------------------------------------
   |
   | The port that expose will try to connect to. If you want to bypass
   | firewalls and have proper SSL encrypted tunnels, make sure to use
   | port 443 and use a reverse proxy for Expose.
   |
   | The free default server is already running on port 443.
   |
   */
   'port' => 3030,

   // ...
```

Running With Docker

#

To run Expose with docker use the included `docker-compose.yaml`. Copy `.env-example` to `.env` and update the configuration.

```
PORT=8080
DOMAIN=example.com
ADMIN_USERNAME=username
ADMIN_PASSWORD=password
```

After updating the environment variables you can start the server:

```
docker-compose up -d
```

Now that your basic expose server is running, let's take a look at how you can add SSL support.

Let us do the heavy lifting.

You want custom domains with automatic SSL, reserved subdomains and the benefits of our global Expose network with blazing fast servers without the hassle of maintaining your own server. Totally understandable.

We've got you covered.