

🔗 Installation process

After copy current version source code go to the root path

```
cd tape/
```

Then run installer with `sudo`

```
sudo ./install.sh
```

What they do:

1. Install main OS packages through `apt` :

- `Nginx`
- `Postgresql`
- `Redis-server` (Key-value storage for user sessions and handling `tape` events)
- `Certbot` (Automatically enable HTTPS on your website with EFF's Certbot)
- `NodeJS`
- `Yarn` (Yarn package manager like `npm`)
- `KnexJS` (Query builder for Postgres, MSSQL, MySQL, MariaDB, SQLite3, Oracle)
- `PM2` (Daemon process manager that will help you manage and keep your application online 24/7)

2. Create database `tape` with user `tape` and grant privileges on database `tape` .

Don't forget after installation process save database user password and put it into `./backend/.env` file in `knex` section.

Go to backend path

```
cd ../backend/
```

Be patient and add your server configuration environment variables:

```
vim .env.production
```

Add new `session.secret` key generated like:

```
openssl rand -base64 48
```

Put database credentials in `knex.production` section, get it from previous step:

```
...,
knex: {
  production: {
    connection: {
      database: "tape",
      user: "tape",
      password: "new_super_secret_password"
    }
  }
},
...
```

Migrate schemas and seed if you need seeding data

```
knex migrate:latest --env production
knex seed:run --specific=002_fixtures.js --env production
```

Run backend server

```
pm2 start
```

Go to frontend and build client

```
cd ../frontend/
```

Before building your bundles you need to change environment variables. Use `.env.production` file:

```
vim .env.production
```

Then run builder and copy build paths to the root server path

```
yarn build
sudo cp -R build/* ~/builds/app/
```

Nginx configuration

To get ssl certificates run:

```
sudo certbot --nginx -d yourdomain.name -d img.yourdomain.name
```

Copy configuration files placed in root `/tape/config` path into `/etc/nginx/sites-available/`. Configure it with your domain names.

Don't forget to use your own root path to the upload directory. It placed in `img.nginx` file and looks like: `root /home/user/tape/frontend/public/uploads;`

Create symlink's:

```
sudo ln -s /etc/nginx/sites-available/default.nginx /etc/nginx/sites-enabled/  
sudo ln -s /etc/nginx/sites-available/img.nginx /etc/nginx/sites-enabled/
```

And restart `Nginx` server:

```
sudo systemctl restart nginx
```