

# OnlyOffice

- [Install OnlyOffice Document Server Ubuntu 18.04](#)
- [Restict Access to OnlyOffice Document Server to Nextcloud Users Only](#)

# Install OnlyOffice Document Server Ubuntu 18.04

**Add the repository containing the up-to-date Node.js package versions:**

```
curl -sL https://deb.nodesource.com/setup_8.x | sudo bash -
```

**Install postgresql:**

```
apt install postgresql
```

**Create Database (Type each command seraparately):**

```
/usr/bin/sudo -i -u postgres psql -c "CREATE DATABASE onlyoffice;"  
/usr/bin/sudo -i -u postgres psql -c "CREATE USER onlyoffice WITH password 'onlyoffice';"  
sudo -i -u postgres psql -c "GRANT ALL privileges ON DATABASE onlyoffice TO onlyoffice;"
```

**Install Redis:**

```
sudo apt-get install redis-server
```

**Install rabbitmq:**

```
sudo apt-get install rabbitmq-server
```

**Install nginx-extras:**

```
sudo apt-get install nginx-extras
```

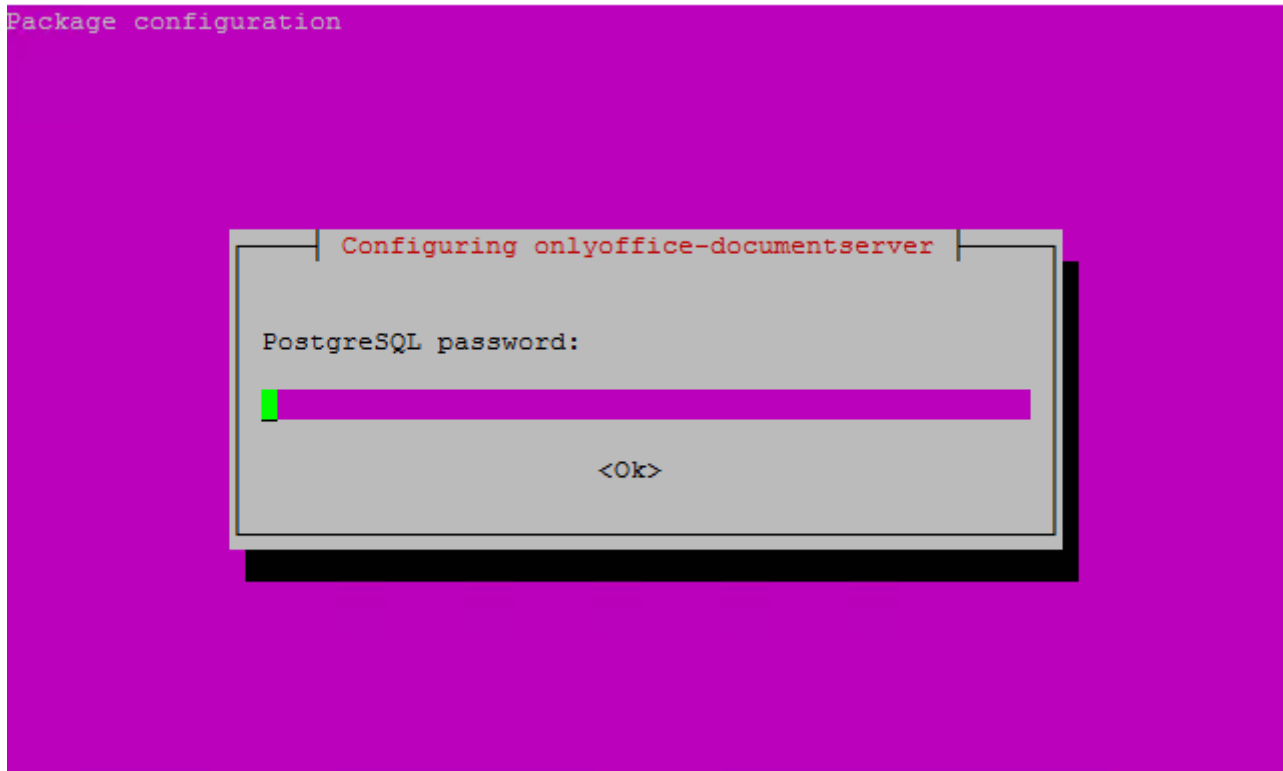
**Install Document Server (Type each command seraparately):**

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys CB2DE8E5  
sudo echo "deb http://download.onlyoffice.com/repo/debian squeeze main" | sudo tee  
/etc/apt/sources.list.d/onlyoffice.list  
sudo apt-get update
```

```
sudo apt-get install onlyoffice-documentserver
```

**Note:** During the installation process, you will be prompted with the following screen (**Figure 1**):

**Figure 1**



Please enter the **onlyoffice** password that you have specified when configuring PostgreSQL.

## Rebuild spellchecker (Ubuntu 18.04):

*Update npm:*

```
sudo npm install -g npm
```

*Stop Document Server services:*

```
sudo supervisorctl stop all
```

*Install the components necessary for the build:*

```
sudo apt-get install build-essential git
```

*Switch to the spellchecker folder:*

```
cd /var/www/onlyoffice/documentserver/server/SpellChecker/
```

*Delete the older build:*

```
sudo mv node_modules/ node_modules_old/
```

*Install the dependencies:*

```
sudo npm install
```

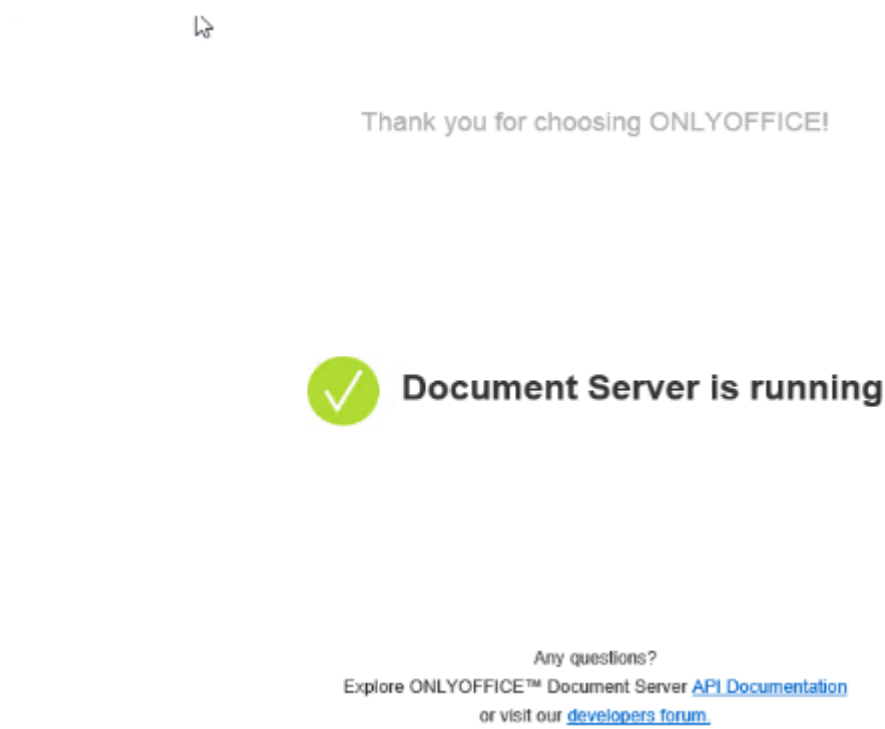
And finally start back Document Server services:

```
sudo supervisorctl start all
```

## Verify that OnlyOffice Document Server is running:

Using a browser browse to the IP address of the OnlyOffice Document server Ex: <http://192.168.xxx.xxx>. You should get the following page (**Figure 2**):

### Figure 2



## Configure OnlyOffice Document Server to use HTTPS:

Stop NGINX service:

```
sudo service nginx stop
```

Create `/etc/nginx/ssl` directory to store certificates and keys:

```
mkdir /etc/nginx/ssl
```

*Obtain the necessary certificates and keys from a 3rd party trusted Certificate Authority before going further*

Place 3rd party certificate and key in `/etc/nginx/ssl` directory as below:

`/etc/nginx/ssl/office.pem`

/etc/nginx/ssl/office.key

Backup existing ds.conf file:

```
cp /etc/onlyoffice/documentserver/nginx/ds.conf /etc/onlyoffice/documentserver/nginx/ds.conf.BACKUP
```

Copy ds-ssl.conf.tmpl to ds.conf:

```
cp /etc/onlyoffice/documentserver/nginx/ds-ssl.conf.tmpl /etc/onlyoffice/documentserver/nginx/ds.conf
```

Edit new ds.conf file and replace all parameters with double curly brackets {{ }} for the actual values:

```
vi /etc/onlyoffice/documentserver/nginx/ds.conf
```

- ssl\_certificate {{SSL\_CERTIFICATE\_PATH}};
- ssl\_certificate\_key {{SSL\_KEY\_PATH}};

So it looks like below:

```
ssl_certificate /etc/nginx/ssl/office.pem;  
ssl_certificate_key /etc/nginx/ssl/office.key;
```

Additionally, edit the **ssl\_protocols** line and remove all but **TLSv1.2** entries so it looks like below:

```
ssl_protocols    TLSv1.2;
```

Start NGINX service:

```
sudo service nginx start
```

## Verify that OnlyOffice Document Server is running on HTTPS:

Using a browser browse to the IP address of the OnlyOffice Document server Ex:  
<https://192.168.xxx.xxx>. You should get the following page (**Figure 3**):

### Figure 3



Thank you for choosing ONLYOFFICE!



**Document Server is running**

Any questions?

Explore ONLYOFFICE™ Document Server [API Documentation](#)  
or visit our [developers forum](#).

# Restrict Access to OnlyOffice Document Server to Nextcloud Users Only

ONLYOFFICE Document Server can use a token generated using the JSON Web Tokens (JWT) standard in order to secure the connection and restrict access to the OnlyOffice Document server which by default is accessible to everyone.

- Edit the **/etc/onlyoffice/documentserver/local.json** file:

```
sudo vi /etc/onlyoffice/documentserver/local.json
```

- Find the following section:

```
"token": {  
  
  "enable": {  
  
    "request": {  
  
      "inbox": false,  
  
      "outbox": false  
  
    },  
  
    "browser": false
```

- Set all the settings from “false” to “true” so it looks like below:

```
"token": {  
  
  "enable": {
```

```
"request": {  
  
  "inbox": true,  
  
  "outbox": true  
  
},  
  
"browser": true
```

- Locate the following section:

```
"secret": {  
  
  "inbox": {  
  
    "string": "secret"  
  
  },  
  
  "outbox": {  
  
    "string": "secret"  
  
  },  
  
  "session": {  
  
    "string": "secret"
```

- Substitute **“secret”** with a secret token of your choosing (ex: ThisIsTheSecret) so it looks like below:

```
"secret": {  
  
  "inbox": {  
  
    "string": "ThisIsTheSecret"  
  
  },
```



```
"outbox": {  
  
  "string": "ThisisTheSecret"  
  
},  
  
"session": {  
  
  "string": "ThisisTheSecret"
```

- Save the file and restart the services:

```
sudo supervisorctl restart all
```

- Go in Nextcloud under **Settings --> ONLYOFFICE** and click on **Advanced server settings**. In the **Secret key** field enter the **THESECRET** token you created earlier and click the Save button (**Figure 1**).

**Figure 1**

### Server settings

ONLYOFFICE Document Service Location specifies the address of the server with th

Document Editing Service address

Secret key (leave blank to disable)

Advanced server settings ▲

Document Editing Service address for internal requests from the server

Server address for internal requests from the Document Editing Service

Save

- Click the “Save” button, you should get the following message on top of the Nextcloud window (Figure 2):

Figure 2

