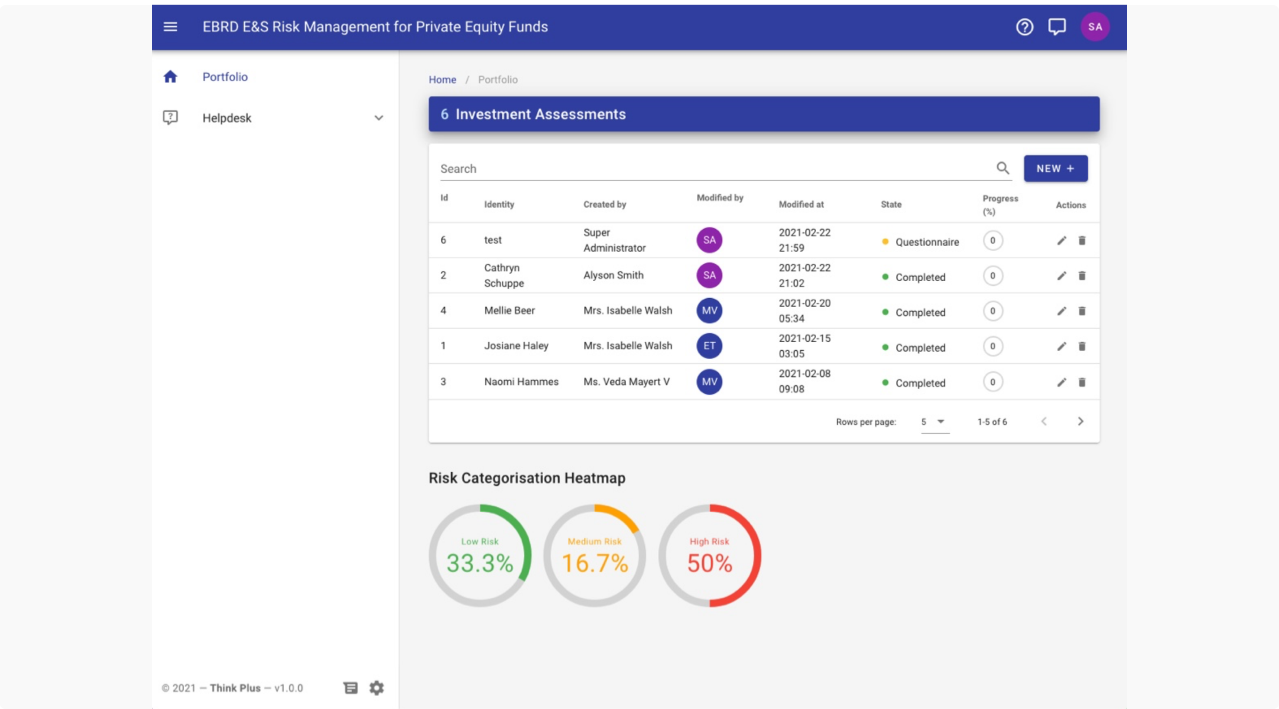
EBRD E&S Risk Management for Private Equity Funds

Technical documentation — v2 (Updated: 02/06/21)

This guide is addressed to the system administrator or IT expert of the financial intermediary who is responsible to install and configure the application on the local intranet infrastructure.

Along with this guide, a compressed file of the application should also be received.

If any further help is required, you should contact support at support@globalsustain.org.



Id	Identity	Created by	Modified by	Modified at	State	Progress (%)	Actions
6	test	Super Administrator	SA	2021-02-22 21:59	Questionnaire	0	
2	Cathryn Schuppe	Alyson Smith	SA	2021-02-22 21:02	Completed	0	
4	Mellie Beer	Mrs. Isabelle Walsh	MV	2021-02-20 05:34	Completed	0	
1	Josiane Haley	Mrs. Isabelle Walsh	ET	2021-02-15 03:05	Completed	0	
3	Naomi Hammes	Ms. Veda Mayert V	MV	2021-02-08 09:08	Completed	0	

1. About the application

The application is a custom web platform build on PHP scripting language. It is served by a web server and in order to access it, a modern browser is required such as Chrome, Firefox, Safari or MS Edge.

> ⚠️ Please note that Microsoft Internet Explorer is not supported.

The app must be installed on a VPS or physical dedicated server on local intranet. Although the app can be installed on any Linux distro, it is recommended to be installed on the latest Ubuntu or CentOS operating systems (preferably an LTS version).

> ⚠️ In order to protect sensitive data and comply with GDPR, the application MUST NOT be installed on any cloud platform or remote server off-premises.

The network topology must allow the app to be reachable by its users by hitting the public domain name of the server on their browser (example: https://ebrdapp.mydomain).

When installed, a super administrator account will be created. This account will be responsible for creating the app users, managing the app settings and handling password resets.

It is not recommended that this account is used by an end-user of the application, unless it is strictly required by the FI administration.

2. Prerequisites

The app should by installed on a physical dedicated machine or virtual machine. It is not recommended to host it in parallel with other services for security reasons.

Although the server configuration is out of the scope of this documentation, because of many different implementations and infrastructure restrictions, we have included instructions to fully provision a fresh Ubuntu 21.04 server to install the app.

You may choose to configure the server on your own and skip the next section but make sure you have installed all of the following prerequisites:

- Nginx 1.18.0 / MySQL 8.0
- PHP 7.4
- BCMath PHP Extension
- Ctype PHP Extension
- Fileinfo PHP Extension
- JSON PHP Extension
- Mbstring PHP Extension
- OpenSSL PHP Extension
- PDO PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension

Minimum server requirements are: 4GB RAM, 4vCPU, 40GB+ HD.

The nginx root path should be the `/public` folder as shown on *section 3*.

PHP configuration should include the following parameters:

```
memory_limit = 2048M
post_max_size = 128M
upload_max_filesize = 128M
```

In case of a firewall, TCP port 80 & 443 should be opened.

It is not mandatory for the server to have internet access. The app is fully capable of running in offline mode. However you might need internet access to configure the server and perform updates.

> ⚠️ The above software versions are just for reference. Although they are the latest stable versions at the time of writing this documentation, you should always try to install the most recent versions that exist to stay secure. We don't encourage the use of outdated software.

3. Server Configuration (Optional)

The following guide will help you configure a fresh Ubuntu 21.04 system to setup a LEMP stack (Linux, Nginx, MySQL, PHP). You must make sure you run the following commands from an account with sudo access. Do not run on root account.

First, update the apt repository and fetch all required prerequisites

```
$ sudo apt-get -y update
$ sudo apt -y install software-properties-common
$ sudo add-apt-repository -y ppa:ondrej/php
$ sudo apt-get -y update
$ sudo apt install unzip curl nano nginx mysql-server
$ sudo apt install -y php7.4 php7.4-fpm php7.4-cli php7.4-common php7.4-bcmath php7.4-json php7.4-mbstring php7.4-xml php7.4-gd php7.4-curl php7.4-zip php7.4-mysql
$ sudo curl -s https://getcomposer.org/installer | php
$ sudo mv composer.phar /usr/local/bin/composer
```

Next, start mysql, and change root password

```
$ sudo systemctl start mysql
$ sudo mysql
```

Enter the following commands, replacing ' PASSWORD ' with your own password. It will also create the database for the application.

```
UPDATE mysql.user SET authentication_string=null WHERE User='root';
flush privileges;
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'PASSWORD';
flush privileges;
CREATE DATABASE ebrd_ens_risk;
exit
```

Add current user to nginx and php-fpm conf, to avoid ownership errors. If you don't know what the current user is you may use the `whoami` command.

```
$ sudo usermod -aG www-data $USER
$ sudo nano /etc/nginx/nginx.conf
```

Change the first line as followed, replacing MY_USER to your username.

```
user MY_USER;
```

Add current user to php-fpm conf

```
$ sudo nano /etc/php/7.4/fpm/pool.d/www.conf
```

Find the following lines and replace MY_USER with your username

```
user = MY_USER
group = MY_USER
```

Assuming you placed the application archive to your home folder (~), the following commands will extract the application archive, move it to the correct location and reset file permissions. Change the version according to your file. Your app path will be **/var/www/ebrdapp.**

```
$ cd ~
$ unzip ebrdapp_v1.1.5.zip -d ebrdapp
$ sudo mv ~/ebrdapp /var/www/ebrdapp
$ sudo find /var/www/ebrdapp -type f -exec chmod 644 {} \;
$ sudo find /var/www/ebrdapp -type d -exec chmod 755 {} \;
$ sudo chown -R $USER:$USER /var/www/ebrdapp
```

In order for the app to be served by nginx, an appropriate configuration file must be created.

```
$ sudo nano /etc/nginx/sites-available/ebrdapp
```

Paste the following configuration, replacing server_domain_or_IP with your public domain that users will use to reach the app. (Example: ebrdapp.mydomain)

```
server {
    listen 80;
    server_name server_domain_or_IP;
    root /var/www/ebrdapp/public;

    client_max_body_size 128M;

    add_header X-Frame-Options "SAMEORIGIN";
    add_header X-XSS-Protection "1; mode=block";
    add_header X-Content-Type-Options "nosniff";

    index index.php;

    charset utf-8;

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location = /favicon.ico { access_log off; log_not_found off; }
    location = /robots.txt  { access_log off; log_not_found off; }

    error_page 404 /index.php;

    location ~ \.php$ {
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;
        include fastcgi_params;
        fastcgi_read_timeout 300;
        fastcgi_param PHP_VALUE "memory_limit = 2048M \n post_max_size = 128M \n
upload_max_filesize = 128M";
    }

    location ~ /\.(?!well-known).* {
        deny all;
    }
}
```

Save the file and exit. (Ctrl-O, Enter, Ctrl-X).

Enable the configuration and check for errors by running the following:

```
$ sudo ln -s /etc/nginx/sites-available/ebrdapp /etc/nginx/sites-enabled/
$ sudo nginx -t
```

Running `sudo nginx -t` should check whether or not there was an issue with the configuration. If the syntax is ok, then a successful message will been shown.

> ⚠️ It is recommend that a TLS certificate is purchased from a trusted root certificate authority and implemented within the web services. See the officially guide here, for instructions.

Finally, restart the php and nginx services.

```
$ sudo systemctl restart php7.4-fpm
$ sudo systemctl restart nginx
```

You are now ready to install the app.

4. Server Hardening (recommended)

4.1. Install Anti-Virus

The application has the ability to upload files on the server. Because of that it is recommended to install an anti-virus application. The anti-virus should have an on-access scanning. Please ensure that the software version and the virus signatures are kept up to date. If required configure Tamper Protection (or require specific user access) in order to modify/stop antivirus services. Ideally utilize a central management console/server for maintaining the antivirus infrastructure.

4.2. Use TLS certificate for secure communication

It is strongly recommended that you use a TLS certificate from a trusted root certificate authority to secure the communication. When configuring the app, specify that you want to use a secure connection over https. After that all communication occurs over an encrypted channel and the application adds the 'secure' attribute to all session cookies or any cookies containing sensitive data.

4.3. Follow best practices

We encourage you to perform a post-install server hardening on the server in line with industry best practice and client standards in order to maximize its security. The type of hardening you carry out depends on the risks in your existing technology, the resources you have available, and the priority for making fixes.

Server Hardening checklist:

- Best Practice HTTP Security Headers
- Reducing the “Attack Surface”
- Components and subsystems
- Updates and vulnerabilities
- Networks and firewalls
- Advanced configuration hardening

5. Installation

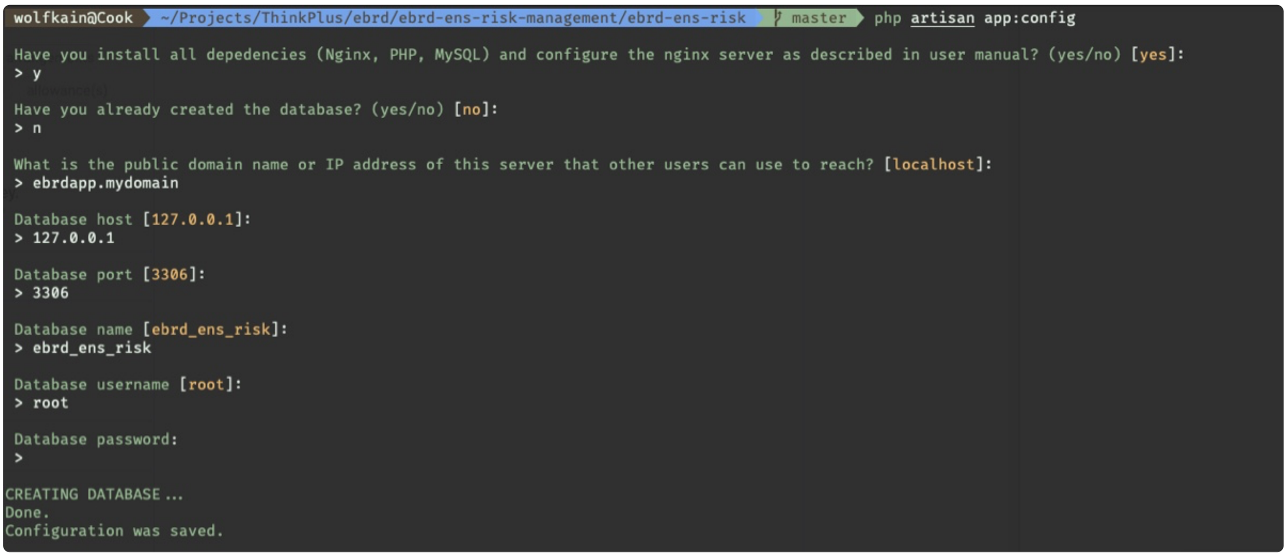
In order to install the app, the server must be properly configured.

Login to the server and cd to your app folder

```
$ cd /var/www/ebrdapp
```

Configure the app by answering a series of questions.

```
$ php artisan app:config
```



```
wolfkain@Cook  ~/Projects/ThinkPlus/ebrd/ebrd-ens-risk-management/ebrd-ens-risk  master  php artisan app:config

Have you install all depedencies (Nginx, PHP, MySQL) and configure the nginx server as described in user manual? (yes/no) [yes]:
> y

Have you already created the database? (yes/no) [no]:
> n

What is the public domain name or IP address of this server that other users can use to reach? [localhost]:
> ebrdapp.mydomain

Database host [127.0.0.1]:
> 127.0.0.1

Database port [3306]:
> 3306

Database name [ebrd_ens_risk]:
> ebrd_ens_risk

Database username [root]:
> root

Database password:
>

CREATING DATABASE ...
Done.
Configuration was saved.
```

Finally, install the app

```
$ php artisan app:install
```

After the installation you will be displayed the default username/password of the super administrator. You should be able to login from any intranet computer by hitting the provided url.

After logging in, you will immediately be prompted to change your password. After doing so, you are ready to customize the app as you wish and create the users.

The app does not use any mail server. You should send the credentials using your internal mail system.

6. Updating

Updating the application is an offline procedure. EBRD will provide the update file whenever a new version is released. Updating the application is a very easy task and can be performed in a matter of seconds.

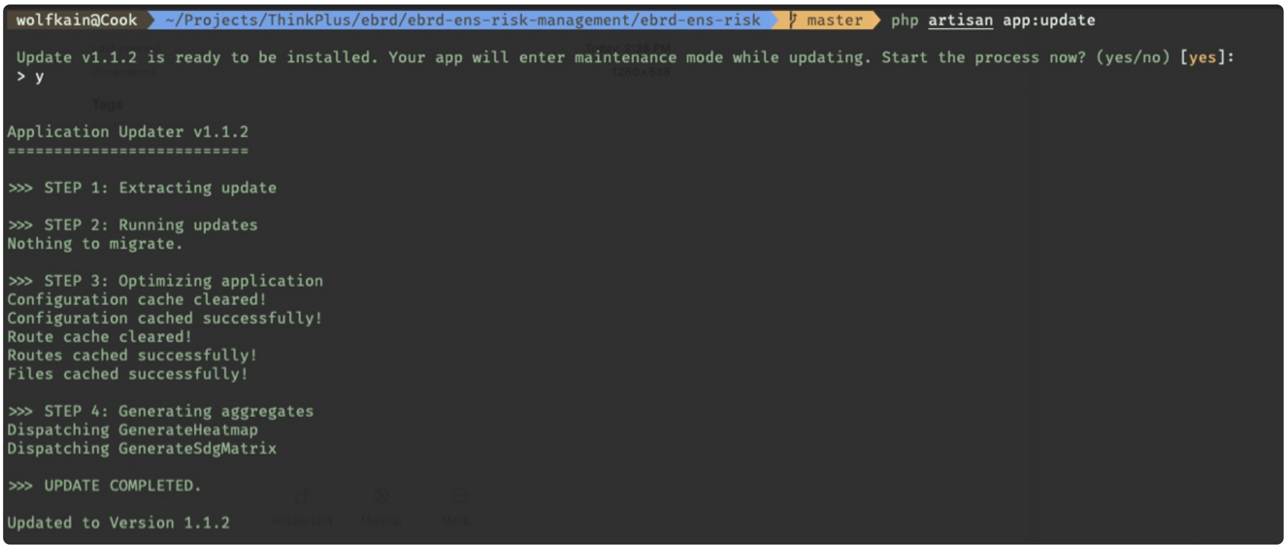
First, login to your server and copy the file to your app folder.

```
$ cd /var/www/ebrdapp
```

The file will be in a form of `ebrdapp_vX.X.X.zip`. (Where X.X.X the future version). **Do NOT rename the file.**

Run the update by running the following command.

```
$ php artisan app:update
```



```
wolfkain@Cook  ~/Projects/ThinkPlus/ebrd/ebrd-ens-risk-management/ebrd-ens-risk  master  php artisan app:update

 Update v1.1.2 is ready to be installed. Your app will enter maintenance mode while updating. Start the process now? (yes/no) [yes]:
 > y

Application Updater v1.1.2
==========================

>>> STEP 1: Extracting update

>>> STEP 2: Running updates
Nothing to migrate.

>>> STEP 3: Optimizing application
Configuration cache cleared!
Configuration cached successfully!
Route cache cleared!
Routes cached successfully!
Files cached successfully!

>>> STEP 4: Generating aggregates
Dispatching GenerateHeatmap
Dispatching GenerateSdgMatrix

>>> UPDATE COMPLETED.

Updated to Version 1.1.2
```

After the the update you will be displayed the new version. The update file will automatically be deleted after successful installation.

> ⚠️ The update takes around 10 seconds. During this process the app will get in maintenance mode and will not allow any users to login. You should warn the users before and after performing the update.

If you need to verify the update, you may use the following command to check for the app version.

```
$ php artisan app
```

In case the update failed see the troubleshooting section.

7. Troubleshooting

In case you get an error during the install or update procedure, a descriptive message will be displayed with directions to follow.

Bellow are a few examples of a broken installation or update.

`Could not create the database` / `Could not migrate the database`

> You have an error on your configuration. Run **`php artisan app:config`** and provide all necessary data again.

`Could not flush session` / `Could not empty storage`

> You have incorrect file permissions or ownership on the `/storage` and `/bootstrap/cache` folders. See *section 4* on how to reset permissions.

`No update file was found.`

> In order to initiate the update you must copy the file provided by EBRD in the root folder of your app path (eg. */var/www/ebrdapp*) without renaming it. If you accidentally renamed the file then restore the original filename by re-downloading it.

If you get an unknown error, it means you have a generic server error. You must verify that all the services are running and check the logs for possible errors. If necessary perform a system reboot. If the problem insists, you may ask for help at supportglobalsustain.org.

· · ·

If you get an error while updating, the app might stay in maintenance mode. First, try to update once again and see if that solves the issue. If you get an error for the second time, you should contact support. In the meantime enter the following commands to restore the app functionality.

```
$ php artisan optimize
$ php artisan up
```

· · ·

7.1. Refreshing Application

Over the time, the application might become slow. Especially with more than 100.000 records. For this reason, aggregates and other time expensive algorithms are cached to reduce the CPU and memory load on the server. The cached data might need to be refreshed once or twice a year. It does refresh automatically after every update. This might also fix some issues of the app and generally it is the first thing to try out when troubleshooting an error.

To refresh the app, login to the server, cd to your app folder and run the refresh command.

```
$ cd /var/www/ebrdapp
$ php artisan app:refresh
```

7.2. Reset Application

If you wish to restore the application to factory settings, meaning deleting all data, settings, files and starting fresh, you may enter the following command after logging in to your server and cd to your app folder. You may for example run this command in case you had an error after your initial installation.

```
$ php artisan app:reset
```

🚨 Warning! **You cannot recover any of your data** after you have reset your app.

7.3. Reset Administrator Password

If you forgot your super administrator password, you may reset your password by logging in to your server, cd to your app folder and run the following command.

```
$ php artisan app:resetAdmin
```

The default password ‘**admin**’ will be restored.