

# PWM Password Management

- [Introduction](#)
- [Installation](#)
- [Configuration](#)

# Introduction

## Introduction

PWM is an open source password self-service application for LDAP directories

It includes:

- LDAP Directory Support
- Change Password module for Self-Service
- Account Activation / First time password assignment
- Password reset
- User registration

Multiple Deployment Options

- Java WAR file (bring your own application server, tested with Apache Tomcat)
- Java single JAR file (bring your own Java VM)
- Docker container

Multiple SSO options

- Basic Authentication
- HTTP header username injection
- Central Authentication Service (CAS)
- OAuth client

REST Server APIs for most functionality

- Password set
- Forgotten password
- Password policy reading
- User attribute updates
- Password policy verification

## Installation / Architecture

The service is installed on portainer.simmy.ch. It is simply the default installation of the docker container.

# Useful links

[Install experience](#)

<https://github.com/pwm-project/pwm>

<https://groups.google.com/g/pwm-general?pli=1>

<https://www.pwm-project.org/pwm/public/reference/>

[Downloads](#)

# Installation

## Introduction

There three different ways for the installation. I choose the docker deployment.

## Requirements

- Debian Linux Server
- pre-installed docker

## Installation

The PWM docker image includes Java and Tomcat. It listens using https on port 8443, and has a volume exposed as /config. You will need to map the /config volume to some type of persistent docker volume for PWM to retain configuration.

## Download the newest version

Goto <https://github.com/pwm-project/pwm/releases>

find and download the most recent .tar file. In my case it was [pwm-docker-image-2.0.6.tar](#).

## Load the docker image

Load your docker image with image name of default pwm/pwm-webapp:

```
docker load --input=pwm-docker-image-v2.0.0.tar
```

## Create file structure

I worked in the root path.

```
mkdir pwm-config
```

This subdirectoy will become very useful, hence there will be all fiels for configuration and debugging puposes.

## Create docker image

Create docker image named mypwm, map to the server's 8443 port, and set the config volume to use the server's local file system /home/user/pwm-config folder (this will be the PWM application path for the container):

```
docker create --name mypwm -p '8443:8443' --mount 'type=bind,source=/root/pwm-config,destination=/config'
pwm/pwm-webapp
```

## Start the mypwm container:

```
docker start mypwm
```

# Configuration

## Introduction

After the installation it is necessary to configure several parameters and options to ensure the system works properly.

## How To change values

The PWM can run in two different modes:

- config read only
- config editable

New registration are only working in the read only mode. If there is the need to change any configuration setting, the PWM config has to be set to editable. To do this, got to the directory /root/pwm-config, edit the file PwmConfiguration.xml and change the following property:

```
<property key="configsEditable">false</property>
```

Fortunately this is the first property of the file.

The key has to be changed from false to true. Save the file and exit the editor. After that open the webpage <https://portal.simmy.org> and you will find on the upper right corner a new menu, that enables you to edit any configuration setting. When you finished editing, save. All changes are written to the file PwmConfiguration.xml. Reopen the file PwmConfiguration.xml and change the property Key configsEditable to true.

In theory all settings can be changed directly in of the file PwmConfiguration.xml. The changes are applied immediately to the application. However, this is not recommended.

## Password policy

It appears that the solution here (in case anyone else ever runs into this) is to change Settings...Password Settings...Password Policy Source to "Local".

<https://groups.google.com/g/pwm-general/c/dQN9irsCZ2w/m/ESp9RLfdCAAJ>

## Valid E-Mail address

The original settings did not allow to enter E-mails with "\_"s. So I had to change the corresponding regex that checks the entered E-Mail address for valid characters. I simply added the "\_" to the list of allowed characters.

```
Regex:^[a-zA-Z0-9 .,']*@$
```

```
Regex:^[a-zA-Z0-9 _.,']*@$
```

## Bug at user registration

For some reasons the Token that is sent out by pwm gets changed by some web handlers or the E-mail software itself. I could at least partially solve it by overriding some defaults directly in the file PwmConfiguration.xml:

```
<setting key="pwm.appProperty.overrides" modifyTime="2024-02-21T16:26:32Z"
syntax="STRING_ARRAY" syntaxVersion="0">
  <label>Settings  ^g  Application  ^g  Application  ^g  App Property Overrides</label>
  <value>security.http.permittedUrlPathCharacters=^[a-zA-Z0-9 _=]*$</value>
</setting>
```

If there is still an error message, just press enter.