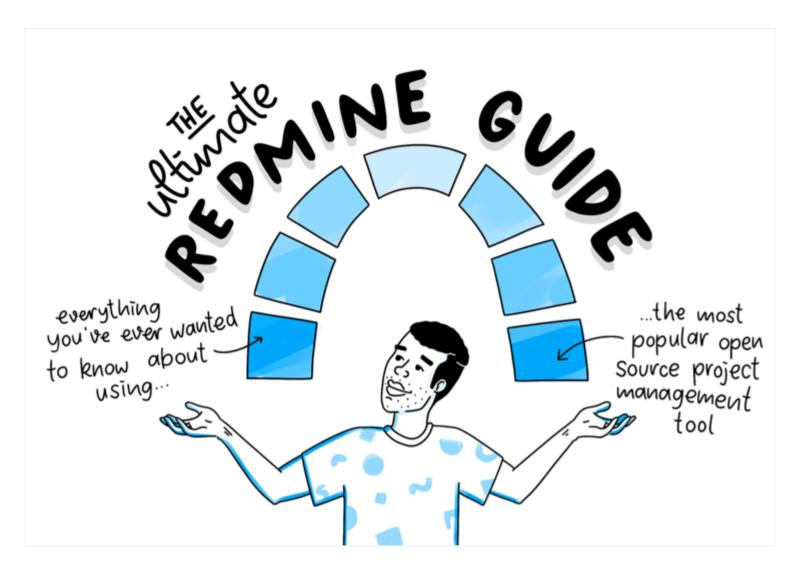


# نصب و راه اندازی سرویس تیکتینگ redmine

در این سند روش نصب و اجرای نرم افزار redmine در محیط های عملیاتی شرح داده شده است.



# ۱- آماده سازی و پیشنیاز ها

## ۱-۱- نصب بسته داکر و داکر کامپوز

برای نصب داکر به صورت **آفلاین**، بستههای زیر را به ترتیب و با استفاده از این سند دانلود و نصب کنید:

برای اجرا و فعالسازی دائمی سرویس داکر دستورات زیر را اجرا کنید:

```
sudo systemctl enable docker
sudo systemctl start docker
```

پس از نصب داکر، با دستورات زیر نام کاربری مورد استفاده در هر ماشین را با دسترسی ویژه به گروه docker اضافه کنید. با انجام این کار دیگر برای اجرای دستورات docker نیاز به نوشتن sudo نیست (در انتها از خط فرمان خارج میشوید).

```
sudo groupadd docker
sudo usermod -aG docker $USER
sudo newgrp
```

توجه: در ادامه فرض میشود که عملیات افزودن کاربر به گروه داکر انجام شده است. در غیر این صورت، تمام دستورات docker را به صورت sudo اجرا کنید. برای نصب docker-compose به صورت **آفلاین** لازم است به شکل زیر عمل کنید:

```
sudo cp docker-compose /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

در صورت دسترسی به اینترنت در محل استقرار سرویس از دستورات زیر برای افزودن gpg key و ریپازیتوری داکر استفاده کنید:

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
   "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
   sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

نصب آنلاین پکیج ها:

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

## ۲-۱- دریافت ایمیج های مورد نیاز

نکته: به علت اینکه redmine به صورت پیشفرض امکان استفاده از certificate برای https را ندارد باید در پشت یک reverse proxy قرار گیرد، در صورتی که در شبکه استقرار چنین سرویسی وجود داشت نیازی به راه اندازی nginx یا haproxy در کنار redmine و دریافت پکیج و انجام تنظیمات مربوط به بخش آن نیز وجود ندارد.

در محیط دارای اینترنت ایمیج های مورد نیاز را دریافت و ذخیره سازی کنید و سپس به محیط air gap منتقل کنید.

نکته: از بین دو گزینه nginx و haproxy میتوانید بسته به نیاز و محیط پیاده سازی یکی را انتخاب کنید.

```
docker pull redmine:5.1.8
docker pull mysql:9.3.0
docker pull haproxy:3.1.8
docker pull nginx:1.28.0
```

## ذخیره سازی و انتقال به محیط آفلاین:

```
docker save -o redmine.tgz redmine:5.1.8
docker save -o mysql.tgz mysql:9.3.0

docker save -o nginx.tgz nginx:1.28.0

docker save -o haproxy.tgz haproxy:3.1.8

scp <file name> <user>@<remote_host_ip>:</destination dir>
```

#### load کردن در محیط استقرار آفلاین:

```
docker load -i redmine.tgz
docker load -i mysql.tgz
docker load -i nginx.tgz
```

docker load -i haproxy.tgz

## ۱-۳- ساخت دایرکتوری های مورد نیاز

در مسیر opt/ دایکرتوری مخصوص redmine و nginx یا haproxy را ایجاد کنید.

mkdir -p /opt/redmine/nginx/certs

mkdir -p /opt/redmine/haproxy/certs

## ۴-۱- ایجاد certificate

#### :nginx

برای راه اندازی در محیط های آفلاین میتوان از self signed certificate استفاده کرد و با دستورات زیر آنها را تولید و در دایرکتوری certs قرار داد.

cd /opt/redmine/nginx/certs
openssl req -newkey rsa:4096 -nodes -keyout ./redmine.key -x509 -days 365 -out ./redmine.crt

**haproxy:** زمانی که haproxy استفاده میشود به علت اینکه نمیتوان key و cert را به صورت جداگانه به آن داد باید هر دو آنها را در یک فایل pem قرار داد.

cd /opt/redmine/haproxy/certs
openssl req -newkey rsa:4096 -nodes -keyout ./redmine.key -x509 -days 365 -out ./redmine.crt
cat redmine.crt redmine.key > redmine.pem

## ۲- نصب و اجرا

## ۱-۲- ساخت docker-compose

برای راه اندازی سرویس های مورد نیاز کافی است فایل docker-compose.yaml را در دایرکتوری اصلی opt/redmine/ با محتوای زیر ایجاد کنید و هر کدام از سرویس های مورد نیاز **nginx** و یا **haproxy** را به بخش مشخص شده در فایل زیر اضافه کنید.

```
version: '3.8'
services:
 # add nginx or haproxy service here
  redmine:
   image: redmine:5.1.8
   container_name: redmine-app
    restart: always
   volumes:
     - 'redmine data:/usr/src/redmine/files'
     - './redmine plugins:/usr/src/redmine/plugins'
     - '/etc/localtime:/etc/localtime:ro'
     - '/etc/timezone:/etc/timezone:ro'
    # if you have local proxy in your network uncomment this part, if not nginx/haproxy service will do
    #ports:
   # - 3000:3000
    environment:
     REDMINE DB MYSQL: mysql
     REDMINE DB PORT: 3306
     REDMINE_DB_DATABASE: ${REDMINE DB DATABASE}
     REDMINE DB USERNAME: ${REDMINE DB USERNAME}
     REDMINE DB PASSWORD: ${REDMINE DB PASSWORD}
     REDMINE SECRET KEY BASE: ${REDMINE SECRET KEY BASE}
    depends on:
      - mysql
   networks:
      - redmine-network
    healthcheck:
     test: ["CMD-SHELL", "wget --no-check-certificate --quiet --method HEAD --output-document /dev/null http://localhost:3000 || exit 1"]
     interval: 30s
     timeout: 10s
     retries: 5
     start period: 60s
 mysql:
   image: mysql:9.3.0
   container name: redmine-db
   restart: always
   volumes:
      mysql data:/var/lib/mysql
    environment:
     MYSQL USER: ${MYSQL USER}
     MYSQL PASSWORD: ${MYSQL PASSWORD}
     MYSQL_ROOT_PASSWORD: ${MYSQL ROOT PASSWORD}
     MYSQL DATABASE: ${MYSQL DATABASE}
   networks:
      - redmine-network
    healthcheck:
     test: ["CMD", "mysqladmin", "ping", "-h", "localhost", "-u", "${MYSQL USER}", "-p${MYSQL PASSWORD}"]
     interval: 30s
     timeout: 10s
```

```
retries: 3
start_period: 30s

volumes:
    mysql_data:
    name: mysql_data
    driver: local
    redmine_data:
    name: redmine_data
    driver: local

networks:
    redmine-network:
    name: redmine-network
    driver: bridge
```

#### ۱-۱-۲ سرویس nginx

```
nginx:
 image: nginx:1.28.0
  container name: redmine-proxy
  restart: always
  ports:
   - "80:80"
   - "8443:443"
  volumes:
   - './nginx/nginx.conf:/etc/nginx/conf.d/nginx.conf'
   - './nginx/certs:/etc/nginx/certs'
   - '/etc/localtime:/etc/localtime:ro'
    - '/etc/timezone:/etc/timezone:ro'
  depends on:
    - redmine
  networks:
   - redmine-network
  healthcheck:
   test: ["CMD", "curl", "-f", "-k", "https://localhost:443"]
   interval: 30s
   timeout: 10s
   retries: 3
   start_period: 10s
```

#### ۲-۱-۲- سرویس haproxy

```
haproxy:
image: haproxy:3.1.8
container_name: redmine-proxy
restart: always
```

```
ports:
 - "80:80"
 - "8443:443"
volumes:
 - './haproxy/haproxy.cfg:/usr/local/etc/haproxy/haproxy.cfg:ro'
 - './haproxy/certs/redmine.pem:/etc/haproxy/certs/redmine.pem:ro'
 - '/etc/localtime:/etc/localtime:ro'
  - '/etc/timezone:/etc/timezone:ro'
depends on:
  - redmine
networks:
  - redmine-network
healthcheck:
 test: ["CMD", "pidof", "haproxy"]
 interval: 30s
 timeout: 10s
 retries: 3
 start period: 10s
```

پس از تکمیل کامپوز فایل مورد نظر برای آن یک فایل .env مشابه مورد زیر ایجاد کنید.

```
### REDMINE
REDMINE_DB_DATABASE=redmine_db
REDMINE_DB_USERNAME=redmine
REDMINE_DB_PASSWORD=<your db pass>
REDMINE_SECRET_KEY_BASE=<your random secret key>

### MYSQL
MYSQL_USER=redmine
MYSQL_PASSWORD=<your db pass>
MYSQL_ROOT_PASSWORD=<yor db root pass>
MYSQL_ROOT_PASSWORD=<dd>
MYSQL_PASSWORD=<dd>
MYSQL_P
```

برای درج مقدار برای REDMINE\_SECRET\_KEY\_BASE میتوان از دستور زیر استفاده کرد.

```
openssl rand -hex 16
```

این secret key برای موارد زیر کاربرد دارد:

- امضا و تایید کوکی های نشست انجام شده بین کاربر و سرویس
  - رمز نگاری داده های رد و بدل شده

■ ایجاد توکن های CSRF (Cross-Site Request Forgery) به منظور محافظت در مقابل درخواست های تایید نشده

## ۲-۲- کانفیگ nginx

در صورتی که در شبکه استقرار سرویس proxy وجود نداشته باشد برای استفاده از ssl/ https حتما باید این سرویس راه اندازی و کانفیگ شود برای این کار در دایرکتوری ایجاد شده opt/redmine/nginx/ فایل کانفیگ **nginx.conf** را با محتوای زیر ایجاد کنید.

```
server {
    listen 80:
    server name localhost;
    return 301 https://$host$request uri; # redirect HTTP to HTTPS
server {
    listen 443 ssl:
    server name localhost;
    ssl certificate /etc/nginx/certs/redmine.crt;
    ssl certificate key /etc/nginx/certs/redmine.key;
    location / {
        proxy pass http://redmine-app:3000;
        proxy set header Host $host:8443; # same as exposed port of host # you can remove port in same network
        proxy set header X-Real-IP $remote addr;
        proxy set header X-Forwarded-For $proxy add x forwarded for;
        proxy set header X-Forwarded-Proto $scheme;
        proxy set header X-Forwarded-Port 8443; # same as exposed port of host # you can delete this line in same network
```

در بخش تنظیمات proxy این فایل باید پورت expose شده از کانتینر به ماشین میزبان سرویس که از طریق آن دسترسی به پنل وب داریم باید ذکر شود، به عنوان مثال پورت 8443 در این کانفیگ استفاده شده است. در صورتی که سیستم کاربر در یک شبکه با سرویس redmine باشد و نیازی به فوروارد کردن پورت نداشته باشد میتوان موارد مشخص شده در کانفیگ را حذف کرد.

## ۳-۲ کانفیگ haproxy

در صورتی که در شبکه استقرار سرویس proxy وجود نداشته باشد برای استفاده از ssl/ https حتما باید این سرویس راه اندازی و کانفیگ شود برای این کار در دایرکتوری ایجاد شده opt/redmine/haproxy/ فایل کانفیگ **haproxy.cfg** را با محتوای زیر ایجاد کنید.

```
global
    log /dev/log local0
    maxconn 4096
    tune.ssl.default-dh-param 2048
defaults
    log global
    mode http
    option httplog
    option dontlognull
    timeout connect 5000
    timeout client 50000
    timeout server 50000
frontend http front
    bind *:80
    redirect scheme https code 301 if !{ ssl fc }
frontend https front
    bind *:443 ssl crt /etc/haproxy/certs/redmine.pem
    option forwardfor
    http-request set-header X-Forwarded-Proto https
    http-request set-header X-Forwarded-Port 8443 # same as exposed port of host # you can delete this line in same network
    default backend redmine back
backend redmine back
    server redmine redmine-app:3000
```

در بخش تنظیمات frontend https\_front این فایل باید پورت expose شده از کانتینر به ماشین میزبان سرویس که از طریق آن دسترسی به پنل وب داریم باید ذکر شود، به عنوان مثال پورت 8443 در این کانفیگ استفاده شده است. در صورتی که سیستم کاربر در یک شبکه با سرویس redmine باشد و نیازی به فوروارد کردن پورت نداشته باشد میتوان خط مشخص شده در کانفیگ را حذف کرد.

## ۳-۲- اجرا

پس از تکمیل فایل های موجود سرویس ها را اجرا میکنیم.

docker-compose up -d

پس از healthy شدن تمامی سرویس ها میتوان با مراجعه به آدرس سرور میزبان و یا آدرس local موجود در سیستم کاربر به نرم افزار دسترسی داشت. برای دسترسی در سیستم کاربر از دستور زیر برای port forward استفاده کنید.

ssh -fNT -L 8443:localhost:8443 user@redmine server ip

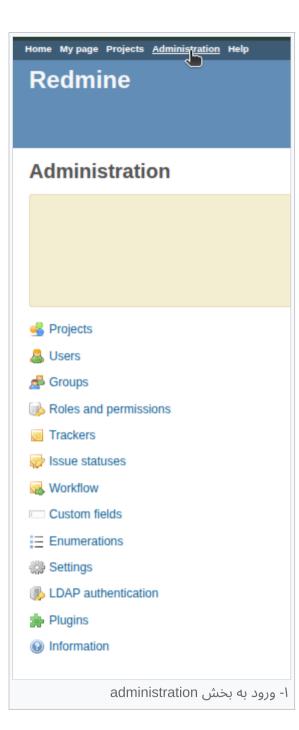
https://localhost:8443/		

# ۲-۲- دسترسی به نرم افزار و تنظیمات وب

سرویس در این آدرس در دسترس است:

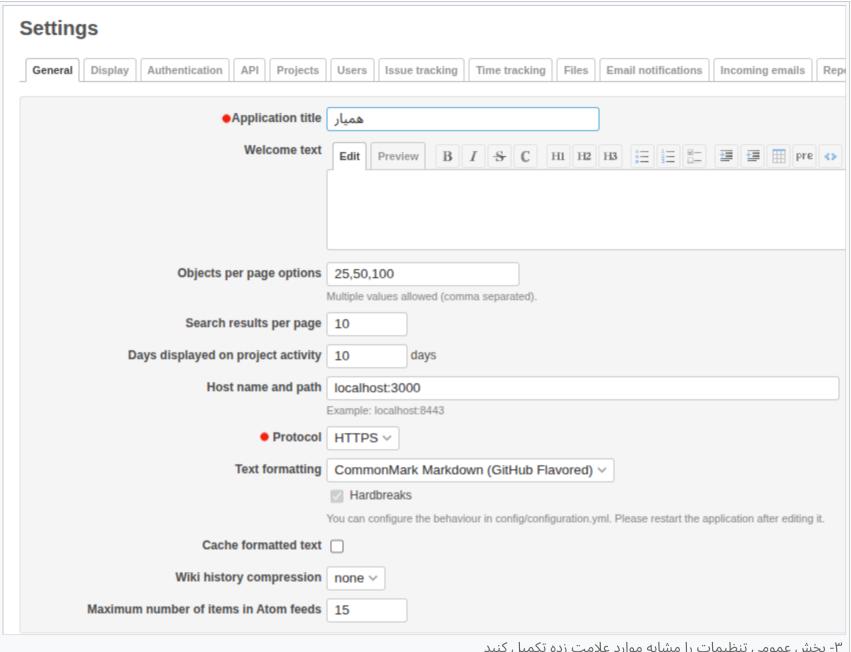
پس از اولین ورود به پنل وب میتوان با کاربر admin و پسورد admin وارد شد و رمز پیشفرض را تغییر داد.

در ورژن 5 به بعد redmine زبان فارسی به صورت کامل پشتیبانی میشود و نیازی به نصب پلاگین وجود ندارد برای اینکار میتوان مشابه تصاویر زیر عمل کرد و زبان پیشفرض و title اصلی برنامه را تغییر داد.

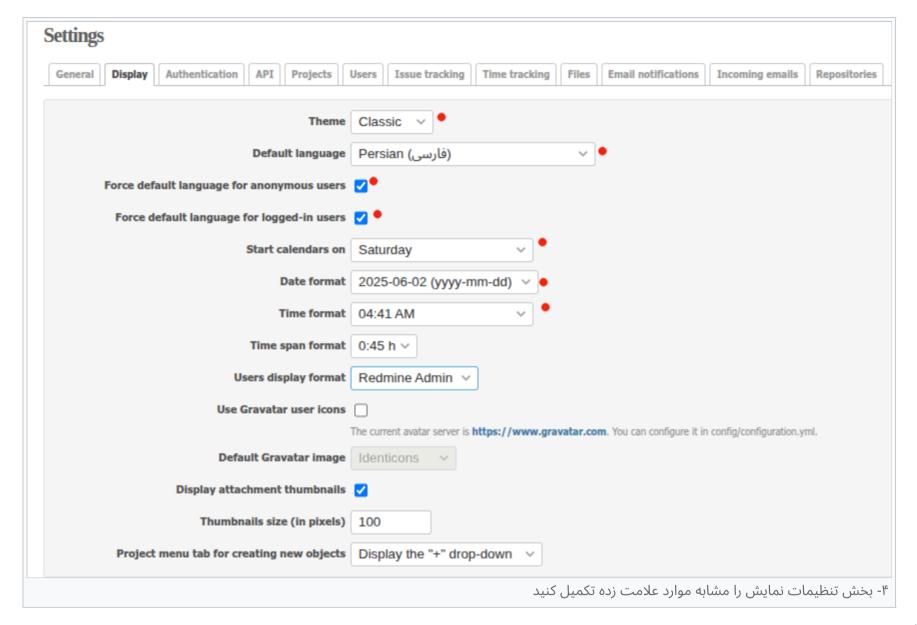


Home My page Projects Administration Help Redmine Administration Projects Users A Groups Roles and permissions Trackers Issue statuses Workflow Custom fields ≡ Enumerations Settings | LDAP authentication Plugins Information

۲- تنظیمات پایه



۳- بخش عمومی تنظیمات را مشابه موارد علامت زده تکمیل کنید



سرویس آماده بهره برداری است!

مشارکتکنندگان: محمد تقوا (99.88), صابر دین پرست (0.13),

بازدید کنندگان: علیرضا حیدری (۱), صابر دین پرست (۳), امیر عباس زاده (۲), محمد تقوا (۲۳)

.