

re50hertz.in

Installation instructions

Info: <https://re50hertz.in>
Author: Manoj Nishad <manoj.n@manikarananalytics.in>
Date: 2017-02-21, revised _____

Introduction

This document explains how re50hertz.in is set up. The instructions are aimed at any competent Linux system administrator.

Server

The application is hosted at E2E Networks in their NOIDA data centre. It runs on a virtual server (plan VPS-HDD-2B). The server configuration is summarised below:

CPU	Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz
RAM	26GB
Disk	440GB
OS	CentOS release 6.8 (Final)
IP	205.147.98.133
Traffic	1000GB/month

Domain

The re50hertz.in domain is registered through Net4India.

SSL Certificate

The application uses a RapidSSL SSL certificate for the **re50hertz.in** domain. This must be renewed periodically.

The private key and CSR (Certificate Signing Request) are generated using the following OpenSSL command:

```
# openssl req -newkey rsa:2048 -nodes -keyout re50.key -out re50.csr  
it will prompt for company information, provide all the details step by step.
```

send the .csr to provider, in return they will send the .cer file. Put the .cer and .key file in the below location.

SSL Certificate store on

```
/etc/nginx/ssl_certificate.cer  
/etc/nginx/re50.key
```

Installation

Application dependencies

1. Setup JAVA Environment

Download Java Archive:

```
# mkdir -p /var/java/  
# cd /var/java/  
# wget --no-cookies --no-check-certificate --header "Cookie: gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie"  
"http://download.oracle.com/otn-pub/java/jdk/7u79-b15/jdk-7u79-linux-x64.tar.gz"  
  
# tar xzf jdk-7u79-linux-x64.tar.gz
```

Install Java & Make link

```
# cd /var/java/jdk1.7.0_79/  
# ln -s /var/java/jdk1.7.0_79/bin/java /usr/bin/java  
# ln -s /var/java/jdk1.7.0_79/bin/java /etc/alternative/java
```

Configuring Environment Variables

Append below given contents in /etc/profile

```
export JAVA_HOME=/var/java/jdk1.7.0_79/  
export PATH=$PATH:/var/java/jdk1.7.0_79/bin
```

Check Installed Java Version

```
# java -version  
java version "1.7.0_79"
```

As We're assuming an apache tomcat instance already running for <https://wind3.50hertz.in>

2. Installing & Configuring Tomcat

Download and Unpack Tomcat 7.0.40

```
# cd /usr/share/  
# wget https://archive.apache.org/dist/tomcat/tomcat-7/v7.0.40/bin/apache-tomcat-7.0.40.tar.gz  
# tar -xvf apache-tomcat-7.0.40.tar.gz  
# mv apache-tomcat-7.0.40 apache-tomcat-solar
```

Configure Tomcat to Run as a Service.

```
# cd /etc/init.d  
# vim solar
```

And here is the script we will use to run tomcat as daemon.

```
#!/bin/bash  
# description: Tomcat Start Stop Restart  
# processname: tomcat  
# chkconfig: 234 20 80  
JAVA_HOME=/var/java/jdk1.7.0_79  
export JAVA_HOME  
servicename=solar  
pidfile=/var/run/$servicename  
PATH=$JAVA_HOME/bin:$PATH  
export PATH  
CATALINA_HOME=/usr/share/apache-tomcat-solar  
case $1 in  
start)  
sh $CATALINA_HOME/bin/startup.sh  
;;  
stop)  
sh $CATALINA_HOME/bin/shutdown.sh  
;;  
restart)
```

```

sh $CATALINA_HOME/bin/startup.sh
;;
#tomcat_pid() { echo "ps -aux | grep $CATALINA_BASE | grep -v grep | tr -s " "|cut -d" " -f2"}
#status)
#   pid=$(tomcat_pid)
#   if [ -n "$pid" ]
#   then
#       echo "Tomcat is running with pid: $pid"
#   else
#       echo "Tomcat is not running"
#   fi
esac
exit 0

```

Change Permission on service script.

```
# chmod 755 solar
```

Deploye war file, taken from Developers/Test Machine

```

SPFS_webapp.war
SOLAR_UI.war
SEL-webapp.war
SEL_UI.war

```

Just need to copy all four war files from Developer machine/ Test Machine to , before that ensure all property files are updated , if not, copy all the property files from production running war, and paste the property files into the appropriate location.

/usr/share/apache-tomcat-solar/webapps/

Adding the service to start on machine startup--

```
# chkconfig solar on
```

Change Port number in **/usr/share/apache-tomcat-solar/conf/server.xml**, because default port already used by **wind3.50hertz.in**

Server port="8006"

SSL port="8080"

Connector port="8010" protocol="AJP/1.3" redirectPort="8080"

Allow port from IPTABLES--

```
# iptables -A INPUT -p tcp --dport 8080 -j ACCEPT
# iptables -A INPUT -p tcp --dport 8443 -j ACCEPT

# /etc/init.d/iptables save
# /etc/init.d/iptables restart
# /etc/inid.d/solar start
```

3. Running Tomcat behind Nginx(Optional)

As an alternative to running Tomcat on port 80, if you have Apache in front of Tomcat, you can use `mod_proxy` to map your domain to your Tomcat application(s) using an Nginx vhost as shown below.

```
# yum install nginx

# nginx -v
nginx version: nginx/1.10.1
```

Allow port from IPTABLES

```
# iptables -A INPUT -p tcp --dport 80 -j ACCEPT
# iptables -A INPUT -p tcp --dport 443 -j ACCEPT
# /etc/init.d/iptables save
# /etc/init.d/iptables restart
```

VHOST with mod_proxy

Append this line in `/etc/nginx/conf.d/default.conf`

```
server {
    listen    80;
    server_name re50hertz.in;
    rewrite ^ https://$http_host$request_uri? permanent;
}

server {
    listen 443;
```

```
server_name re50hertz.in;
ssl on;
ssl_certificate /etc/nginx/ssl_certificate.cer;
ssl_certificate_key /etc/nginx/re50.key;
location / {
    proxy_set_header X-Forwarded-Host $host;
    proxy_set_header X-Forwarded-Server $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_pass https://re50hertz.in:8080/SOLAR_UI/;
}
```

Restart the Service

```
# /etc/init.d/nginx restart
# /etc/init.d/solar restart
```

Now, Go to web browser and type <https://re50hertz.in> in the URL bar

You will be able to see login page