

# 災害リスク情報クリアリングハウス

## インストールマニュアル

Version 1.2

2015/01/15

独立行政法人 防災科学技術研究所

変更履歴

Version	日付	変更内容
1.0	2013/8/23	初版作成
1.1	2014/1/17	他ドキュメントのバージョンアップに伴い版数のみ更新
1.1.1	2014/6/13	他ドキュメントのバージョンアップに伴い版数のみ更新
1.2	2015/1/15	動作環境でインストールするソフトウェアを現時点の最新バージョンに変更

## 目次

1. インストールの前に.....	4
1.1. サーバ環境 .....	4
1.2. ネットワーク設定 .....	5
2. 動作環境のインストール .....	6
2.1. HTTP サーバ(httpd)のインストール .....	6
2.2. Java のインストール.....	7
2.3. Apache Tomcat のインストール.....	8
2.4. PostgreSQL のインストール.....	10
3. 災害リスク情報クリアリングハウスのインストール.....	12
3.1. パッケージのダウンロード.....	12
3.2. パッケージの展開.....	12
3.3. データベースの作成 .....	12
3.4. 実行環境の設定 .....	12
3.5. Tomcat の再起動と動作チェック .....	14

## 1. インストールの前に

災害リスク情報クリアリングハウスを動作させるためには、以下のソフトウェア環境とそれが動作する十分なハードウェアスペックが必要です。

以降で説明するコマンド実行等の操作は、特にユーザも指定が無い限り全て root で実行します。

### 1.1. サーバー環境

#### ● ソフトウェア環境

名称	ソフトウェア詳細
OS	Red Hat Enterprise Linux ES 6 (推奨) CentOS 6 (推奨) Red Hat Enterprise Linux ES 5 CentOS 5 (※64bitOS 推奨)
HTTP サーバ	Apache 2.2 以降
DB サーバ	PostgreSQL 9.2 以降
Java 実行環境	Oracle Java JDK7 以降
Servlet 実行環境	Apache tomcat 7.0 以降

#### ● サーバスペック

名称	要求スペック
CPU	Intel Core i7 または同等の CPU 以上
メモリ	4G byte 以上
ハードディスク	50G byte 以上

#### ● ネットワーク環境

以下の URL を利用していないことを確認して下さい。

```
http://サーバ名/CatalogueService
```

本書に従ってインストールした災害リスク情報クリアリングハウスの URL は以下の通りとなります。

API	http://サーバ名/CatalogueService/servlet/NIEDCatalogueService
Web インタフェース	http://サーバ名/CatalogueService/drigw/top.jsp

/etc/hosts に localhost の設定およびサーバ名(FQDN)が設定されていることを確認します。

設定例: サーバの FQDN が server.domain.com の場合

```
127.0.0.1 localhost
127.0.0.1 server.domain.com
```

環境変数 HOSTNAME にサーバのドメイン名(FQDN)が設定されていることを確認します。

```
echo $HOSTNAME
```

設定を変更する場合は以下を実行します。

環境変数に設定

```
export HOSTNAME=server.domain.com
```

/etc/sysconfig/network を編集

```
NETWORKING=yes  
NETWORKING_IPV6=no  
HOSTNAME=server.domain.com
```

## 1.2. ネットワーク設定

- SELinux の設定

以下を実行して SELinux を無効にします。

```
/usr/sbin/setenforce 0
```

再起動後も無効になるように /etc/selinux/config を編集します。

```
# This file controls the state of SELinux on the system.  
# SELINUX= can take one of these three values:  
# enforcing - SELinux security policy is enforced.  
# permissive - SELinux prints warnings instead of enforcing.  
# disabled - SELinux is fully disabled.  
SELINUX=disabled  
# SELINUXTYPE= type of policy in use. Possible values are:  
# targeted - Only targeted network daemons are protected.  
# strict - Full SELinux protection.  
SELINUXTYPE=targeted
```

- iptables の設定

ポート 80 が開放されていない場合は、以下を実行してポートを開放します。

```
/sbin/iptables -I INPUT -p tcp -m tcp --dport 80 --syn -j ACCEPT  
/etc/init.d/iptables save
```

## 2. 動作環境のインストール

### 2.1. HTTP サーバ (httpd) のインストール

コンソールで以下のコマンドを実行します。

```
yum install httpd
```

/etc/httpd/conf.d/clearinghouse.conf を以下の内容で作成します。

```
ProxyPassMatch /CatalogueService/(.*) ajp://localhost:8009/CatalogueService/$1
```

HTTP サーバが OS 再起動時に自動的に実行されるように以下のコマンドを実行します。

```
chkconfig httpd on
```

HTTP サーバを起動します。

```
/etc/init.d/httpd start
```

動作チェック

```
http://サーバ名/
```

上記 URL を Web ブラウザで表示して Apache 2 Test Page が表示されることを確認します。

## 2.2. Java のインストール

Java SE 7 の最新版パッケージを以下のサイトからダウンロードします。

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

- jdk-7u\*\*-linux-x64.tar.gz - 64bit OS の場合
- jdk-7u\*\*-linux-i586.tar.gz - 32bit OS の場合

(2015/1/15 時点の最新は 7u72 です。以下 jdk-7u72-linux-x64.tar.gz を例に説明します)

パッケージを展開し、所有者を root に変更します。

```
tar xzf jdk-7u72-linux-x64.tar.gz
chown -R root.root jdk1.7.0_72
```

展開したパッケージを /usr/local/java に配置します。

```
mkdir -p /usr/local/java
mv jdk1.7.0_72 /usr/local/java
```

動作チェック

```
/usr/local/java/jdk1.7.0_72/bin/java -version
```

上記コマンドを実行して以下のように表示されることを確認します。

```
java version "1.7.0_72"
Java(TM) SE Runtime Environment (build 1.7.0_72-b14)
Java HotSpot(TM) 64-Bit Server VM (build 24.72-b04, mixed mode)
```

## 2.3. Apache Tomcat のインストール

Apache Tomcat 7 の最新版を以下のサイトからダウンロードします。

<http://tomcat.apache.org/download-70.cgi>

- apache-tomcat-7.0.\*\*.tar.gz - (Binary Distributios > Core > tar.gz を選択)

(2015/1/15 時点の最新は 7.0.57 です。以下 apache-tomcat-7.0.57.tar.gz を例に説明します)

パッケージを展開し、所有者を nobody に変更します。

```
tar xzf apache-tomcat-7.0.57.tar.gz
chown -R nobody.nobody apache-tomcat-7.0.57
```

展開したパッケージを /usr/local に配置します。

```
mv apache-tomcat-7.0.57 /usr/local
```

jsvc (Tomcat をデーモンとして実行するプログラム) をビルドします。

※ jsvc のビルドには gcc および make が必要です。これらが未インストールの場合は以下のコマンドでインストールしてください。

```
yum install gcc
yum install libcap-devel
yum install make
```

```
cd /usr/local/apache-tomcat-7.0.57/bin
tar xzf commons-daemon-native.tar.gz
cd commons-daemon-1.0.15-native-src/unix
./configure --with-java=/usr/local/java/jdk1.7.0_72
make
mv jsvc ../../
```

Tomcat 起動スクリプトを配置します。

```
cd /usr/local/apache-tomcat-7.0.57/bin
cp -p daemon.sh /etc/init.d/tomcat7
```

Tomcat 起動スクリプト (/etc/init.d/tomcat7) を編集します。

```
vi /etc/init.d/tomcat7
```

1 行目 (#/bin/sh) の次に以下の 5 行を追加します。

```
# chkconfig: - 85 15
JAVA_HOME=/usr/local/java/jdk1.7.0_72
TOMCAT_USER=nobody
CATALINA_HOME=/usr/local/apache-tomcat-7.0.57
LANG=ja_JP.UTF-8
```

※ JAVA\_HOME には「2.2Java のインストール」でインストールしたディレクトリを指定します。

Tomcat 実行環境設定 スクリプト(/usr/local/apache-tomcat-7.0.57/bin/setenv.sh)を作成します。

```
vi /usr/local/apache-tomcat-7.0.57/bin/setenv.sh
```

以下の内容で作成します。

```
JAVA_OPTS="-Xms1024m -XX:PermSize=512m"
CATALINA_OPTS="-Dorg.apache.jasper.compiler.Parser.STRICT_QUOTE_ESCAPING=false"
```

Tomcat が OS 再起動時に自動的に実行されるように以下のコマンドを実行します。

```
chkconfig --add tomcat7
chkconfig tomcat7 on
```

Tomcat を起動します。

```
/etc/init.d/tomcat7 start
```

動作チェック

```
http://サーバ名/CatalogueService/
```

上記 URL を Web ブラウザで表示して Tomcat によるエラーページ(HTTP ステータス 404)が表示されることを確認します。

## 2.4. PostgreSQL のインストール

以下の URL から postgres をインストールするためのリポジトリをダウンロードします。

<http://yum.postgresql.org/repopackages.php>

※ OS に適合した PostgreSQL9.2 のリポジトリをダウンロードしてください。

ダウンロードしたリポジトリをインストールします (CentOS6、64bit の例、2015/1/15 時点)。

```
rpm -ivh pgdg-centos92-9.2-6.noarch.rpm
```

PostgreSQL9.2 をインストールします。(9.2 の最新版がインストールされます)

```
yum install postgresql92 postgresql92-server postgresql92-jdbc
```

データベースを初期化します。

```
su -l postgres -c "/usr/pgsql-9.2/bin/initdb -E UTF-8 --no-locale -D /var/lib/pgsql/9.2/data"
```

動作環境 (/var/lib/pgsql/9.2/data/postgresql.conf) の設定を行います。

```
cd /var/lib/pgsql/9.2/data/  
cp -p postgresql.conf postgresql.conf.orig #オリジナルファイルのバックアップ作成  
vi /var/lib/pgsql/9.2/data/postgresql.conf
```

以下の行の値を変更します。

```
shared_buffers = 1GB  
work_mem = 32MB
```

※ shared\_buffers はメモリの 25%程度を設定します。上記は OS のメモリが 4Gbyte の場合の例です。

※ work\_mem はコメント行となっているのでコメント(行頭の#)を外してから値を変更します。

PostgreSQL9.2 が OS 再起動時に自動的に実行されるように以下のコマンドを実行します。

```
chkconfig postgresql-9.2 on
```

PostgreSQL9.2 サーバを起動します。

```
/etc/init.d/postgresql-9.2 start
```

動作チェック

```
psql -l -U postgres
```

上記コマンドを実行して以下のように表示されることを確認します。(一部表示を省略しています)

## データベース一覧

名前	所有者	エンコーディング	照合順序	Ctype(変換演算子)	アクセス権
postgres	postgres	UTF8	C	C	
template0	postgres	UTF8	C	C	
template1	postgres	UTF8	C	C	

(3 行)

### 3. 災害リスク情報クリアリングハウスのインストール

#### 3.1. パッケージのダウンロード

以下の URL から災害リスク情報クリアリングハウスのパッケージをダウンロードします。

<http://ecom-plat.jp/>

パッケージファイル: CatalogueService-install-1.0.tar.gz

#### 3.2. パッケージの展開

ダウンロードしたパッケージを Tomcat インストールディレクトリの webapps に展開します。

(Tomcat インストールディレクトリ: /usr/local/apache-tomcat-7.0.57 の場合)

```
cd /usr/local/apache-tomcat-7.0.57/webapps
tar zxf [パッケージダウンロードディレクトリ]/CatalogueService-install-1.0.tar.gz
```

パッケージは以下のディレクトリに展開されます。

```
/usr/local/apache-tomcat-7.0.57/webapps/CatalogueService
```

#### 3.3. データベースの作成

展開したパッケージに含まれるスクリプトを実行してデータベースを作成します。

```
cd /usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF/db
sh createdb.sh
```

#### 3.4. 実行環境の設定

アプリケーション実行環境設定ファイル(web.xml)を実際の環境に合わせて変更します。

```
cd /usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF
vi web.xml
```

以下の下線部を実際の Tomcat インストールディレクトリに合わせて変更します。

```

<init-param>
  <param-name>jp.co.fal.nied.csw.ThumbnailDir</param-name>
  <param-value>/usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/thumbnails/</param-value>
</init-param>
<init-param>
  <param-name>jp.co.fal.nied.csw.ThumbnailURL</param-name>
  <param-value>http://localhost/CatalogueService/thumbnails/</param-value>
</init-param>
<init-param>
  <param-name>jp.co.fal.nied.csw.MapdataDir</param-name>
  <param-value>/usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/mapdata/</param-value>
</init-param>
<init-param>
  <param-name>jp.co.fal.nied.csw.MapdataURL</param-name>
  <param-value>http://localhost/CatalogueService/mapdata/</param-value>
</init-param>

```

「サイトに関するお問い合わせ」をメールで送信するための設定を行います。

```

cd /usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF/classes/
vi MailTransfer.properties

```

以下の項目に対して値を設定します。

```

mail.smtp.host= (smtp サーバのホスト名)
mail.host= (上記と同じ)
mail.sendto_address= (メール送信先のメールアドレス)
mail.from_address= (送信元のメールアドレス)
mail.from_name= (送信者の名称)
mail.subject= (メールの件名)

```

ログの出力先の設定を行います(アプリケーションログの出力先)。

```

cd /usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF/classes/
vi log4j.xml

```

以下の下線部を実際の Tomcat インストールディレクトリに合わせて変更します。

```

<param name="File" value="/usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF/
logs/csw.log" />

```

続けてもう1つのログの出力先の設定を行います(Tomcat ログの出力先)。

```

vi logging.properties

```

以下の下線部を実際の Tomcat インストールディレクトリに合わせて変更します。

```
org.apache.juli.FileHandler.directory = /usr/local/apache-tomcat-7.0.57/webapps/CatalogueService/WEB-INF/logs
```

### 3.5. Tomcat の再起動と動作チェック

インストールした災害リスク情報クリアリングハウスを実行するために Tomcat を再起動します。

```
/etc/init.d/tomcat7 stop  
/etc/init.d/tomcat7 start
```

本書に従ってインストールした災害リスク情報クリアリングハウスの URL は以下の通りとなります。

API	http://サーバ名/CatalogueService/servlet/NIEDCatalogueService
Web インタフェース	http://サーバ名/CatalogueService/dr1gw/top.jsp

API は e コミマップと相互運用gサーバーと連携する際に必要な URL です。設定方法はそれぞれのシステムのドキュメントをご覧ください。

上記 Web インタフェースの URL をブラウザに入力して動作チェックを行います。

以下の内容が表示されていればインストールは成功です。



<b>新着データ</b>	もっと見る
新着データはありません	

<b>注目のキーワード</b> 詳細なランキング
注目のキーワードはまだありません

<b>注目のデータ</b>	詳細なランキング
注目のデータはまだありません	

<b>おすすめ</b>
おすすめのデータはまだありません