

Configurator

1	FOREWORD	3
2	JSPSERVLETCONFIG	3
2.1	CONFIGURATION.....	3
2.2	DISPLAY.....	3
2.2.1	<i>ID</i>	4
2.2.2	<i>Keystore</i>	4
2.2.3	<i>Password</i>	5
2.2.4	<i>CA certificate URL</i>	5
2.2.5	<i>CRL certificate URL</i>	5
2.2.6	<i>CA user</i>	5
2.2.7	<i>CA password</i>	5
2.2.8	<i>CRL user</i>	5
2.2.9	<i>CRL password</i>	6
2.2.10	<i>Trace</i>	6
2.2.11	<i>Statistics</i>	6
2.2.12	<i>Resource expiration time</i>	6
2.2.13	<i>CRL scan period</i>	6
2.2.14	<i>Cache path</i>	6
2.2.15	<i>Log file</i>	6
2.2.16	<i>Remote location property file</i>	6
2.2.17	<i>All permission policy file</i>	7
2.2.18	<i>Default policy file</i>	7
2.2.19	<i>ServletLog name</i>	7
2.2.20	<i>ServletStat name</i>	7
2.2.21	<i>Web.xml directory</i>	7
2.2.22	<i>JSPservlet URL</i>	7
2.3	COMMANDS.....	8
2.3.1	<i>Check</i>	8
2.3.2	<i>Generate</i>	10
2.3.3	<i>Set</i>	14
2.4	PACKAGING.....	14
2.4.1	<i>Source</i>	14
2.4.2	<i>Class</i>	14
3	PUBLISHCONFIG	14
3.1	CONFIGURATION.....	14
3.2	DISPLAY.....	14
3.2.1	<i>Repository directory</i>	15
3.2.2	<i>Repository URL</i>	15
3.2.3	<i>PageBoxURLs serialized file</i>	15
3.2.4	<i>Max archive size</i>	16
3.2.5	<i>Log file</i>	16
3.2.6	<i>Trace</i>	16
3.2.7	<i>Web.xml directory</i>	16
3.2.8	<i>PublisherServer directory</i>	16
3.3	COMMANDS.....	16
3.3.1	<i>Check</i>	16
3.3.2	<i>Generate</i>	17
3.3.3	<i>Set</i>	19
3.4	PACKAGING.....	19
3.4.1	<i>Source</i>	19
3.4.2	<i>Class</i>	19

1 Foreword

The goal of Configurator is to assist the configuration of PageBox and PublisherServer.

From a remote location it allows to:

- ?? Check configuration parameters on the machines where PageBox and PublisherServer will be deployed
- ?? Generate PageBox and PublisherServer web.xml
- ?? Install PageBox and PublisherServer

It can be used to configure PageBox and PublisherServer either on an Intranet or on ASPs.

Though PageBox and PublisherServer have only few extensively documented parameters, we develop Configurator:

1. To help users to create their first configuration
2. Because some parameters are absolute paths likely to change for instance from an ASP to another
3. To simplify the setting of PageBox constellations

Configurator is made of two servlets:

- ?? JSPservletConfig, responsible to configure PageBox
- ?? PublishConfig, responsible to configure PublisherServer

Configurator principle is to install the bootstrap servlets on target machines to set up PageBox and PublisherServer. For the moment it supports only application server (Tomcat and Resin) version.

2 JSPservletConfig

2.1 Configuration

JSPservletConfig has no initialization parameter. It can be configured in web.xml as below:

```
<servlet>
  <servlet-name>JSPservletConfig</servlet-name>
  <servlet-class>configurator.JSPservletConfig</servlet-class>
</servlet>

<servlet-mapping>
  <servlet-name>JSPservletConfig</servlet-name>
  <url-pattern>/JSPservletConfig</url-pattern>
</servlet-mapping>
```

With Tomcat 3.1 or 3.2, it can simply be copied in the servlet directory and invoked through org.apache.tomcat.servlets.InvokerServlet with a URL <http://myhostport/servlet/JSPservletConfig>.

2.2 Display

When you call JSPservletConfig, the first time, you are displayed the form below with default values:

Address <http://localhost:8080/pagebox/JSPServletConfig>

ID *

Keystore keystore

Password keystorePassword

CA certificate URL

CRL certificate URL

CA user

CA password

CRL user

CRL password

Trace

Statistics

Resource expiration time 30

CRL scan period 30

Cache path *

Log file E:\Jakarta3.2.1\jakarta-tomcat-3.2.1\bin\JSPServlet.log

Remote location property file E:\Jakarta3.2.1\jakarta-tomcat-3.2.1\bin\remoteLocation.prop

AllPermission policy file E:\Jakarta3.2.1\jakarta-tomcat-3.2.1\bin\allPermission.policy

Default policy file E:\Jakarta3.2.1\jakarta-tomcat-3.2.1\bin\default.policy

ServletLog name ServletLog

ServletStat name ServletStat

The following two parameters are used only if you click set and want to upload JSPServlet.

web.xml directory *

JSPServlet URL *

Figure 1: JSPServletConfig

2.2.1 ID

Unique JSPServlet identifier.

Web.xml initialization parameter for JSPServlet: ID.
Default value: none. Mandatory.

2.2.2 Keystore

keystore is the name of the key store in cachePath directory, for instance “keystore” but not “mydir/keystore” or “mydir/keystore”.

If *keystore* is set, when JSPServletPkg downloads an archive, it tries

- ?? To download a certificate from the same location as the archive and named *archive.cer*. If it finds, it adds the certificate to *cachePath/keystore*, which has to be in Sun JKS format with an *archive* alias
- ?? To download a permission file from the same location as the archive and named *archive.policy*. This file should only contain permission entries. JSPServletPkg adds a keystore line, builds the appropriate grant line and stores it in *cachePath/archive.policy* in order to implement a sandbox with the permissions requested by the provider.

Web.xml initialization parameter for JSPServlet: keystore.
Default value: keystore.

2.2.3 Password

Password JSPServletPkg uses to access the keystore.

Web.xml initialization parameter for JSPServlet: keystorePassword.
Default value: keystorePassword.

2.2.4 CA certificate URL

When JSPServletPkg implements sandboxes, it processes signed archives and retrieve classes certificate chain. If CAURL is set, it connects to this URL and expects to retrieve a Certificate Authority certificate used in the classes certificate chain.

Web.xml initialization parameter for JSPServlet: CAURL.
Default value: none.

2.2.5 CRL certificate URL

When JSPServletPkg implements sandboxes, it processes signed archives and retrieve classes certificates.
If CRLURL is set, it connects to this URL and expects to retrieve a Certificate Revocation List (CRL) used to check if a class certificate is revoked.

Web.xml initialization parameter for JSPServlet: CRLURL.
Default value: none.

2.2.6 CA user

Principal used to connect to the Directory server to retrieve CAURL.

Web.xml initialization parameter for JSPServlet: CALDAPuser.
Default value: none.

2.2.7 CA password

Password used to connect to the Directory server to retrieve CAURL.

Web.xml initialization parameter for JSPServlet: CALDAPpasswd.
Default value: none.

2.2.8 CRL user

Principal used to connect to the Directory server to retrieve CRLURL.

Web.xml initialization parameter for JSPServlet: CRLLDAPuser.
Default value: none.

2.2.9 CRL password

Password used to connect to the Directory server to retrieve CRLURL.

Web.xml initialization parameter for JSPServlet: CRLLDAPpasswd.

Default value: none.

2.2.10 Trace

Tells if JSPServlet must write diagnostic messages.

Web.xml initialization parameter for JSPServlet: toTrace.

Default value: set.

2.2.11 Statistics

Tells if JSPServlet must record statistics.

Web.xml initialization parameter for JSPServlet: toStat.

Default value: set.

2.2.12 Resource expiration time

You set this parameter to minimize the round trip number between the browser and the server. JSPServlet sets the *Expire* header field of static content (content with an extension different of *class*). It computes the *Expire* as *current_time + expiration*. *expiration* unit is second.

Web.xml initialization parameter for JSPServlet: expiration.

Default value: 30.

2.2.13 CRL scan period

Defines how often JSPServlet connect to check for CRL updates in seconds.

Web.xml initialization parameter for JSPServlet: CRLperiod.

Default value: 30.

2.2.14 Cache path

Location where presentation archives are locally stored after been retrieved from remote location.

Web.xml initialization parameter for JSPServlet: CachePath.

Default value: current path in JSPServletConfig context.

This default is a clue, not an appropriate value.

2.2.15 Log file

Tells where JSPServlet must write diagnostic messages.

Web.xml initialization parameter for JSPServlet: logFile.

Default value: Cache path + File.separator + JSPServlet.log.

If you set Cache path to a value different of current path and click on Check, then Log file is updated.

2.2.16 Remote location property file

Location of a property file containing presentation archive names and associated URLs.

Web.xml initialization parameter for JSPservlet: `remoteLocations`.
Default value: `Cache path + File.separator + remoteLocation.properties`.

If you set Cache path to a value different of current path and click on Check, then Remote location property file is updated.

2.2.17 All permission policy file

If it is set, *allPermissionPolicy* is the path to a policy file with syntax conforming to the Java 2 security specification. If *defaultPolicy* is also set it means:

1. JSPservletPkg will implement sandboxes. So every archive will run with the permissions defined either in *cachePath/archive.policy* or in *cachePath/java.policy* where:
 - *cachePath* is the *cachePath* initialization parameter value
 - *archive* is the archive name without suffix
2. The Java server itself will run with the permission described in *allPermissionPolicy*.

Web.xml initialization parameter for JSPservlet: `allPermissionPolicy`.
Default value: `Cache path + File.separator + allPermission.policy`.

2.2.18 Default policy file

If it is set, *defaultPolicy* is the path to a policy file with syntax conforming to the Java 2 security specification. If *allPermissionPolicy* is also set it means:

1. JSPservletPkg will implement sandboxes. So every archive will run with the permissions defined either in *cachePath/archive.policy* or in *cachePath/java.policy* where:
 - *cachePath* is the *cachePath* initialization parameter value
 - *archive* is the archive name without suffixIf no policy applies to the archive, then *defaultPolicy* is used
2. The Java server itself will run with the permission described in *allPermissionPolicy*.

Web.xml initialization parameter for JSPservlet: `defaultPolicy`.
Default value: `Cache path + File.separator + default.policy`.

2.2.19 ServletLog name

Sets the name JSPservlet log display will be called with (<servlet -mapping> tag in web.xml).
Default value: `ServletLog`.

2.2.20 ServletStat name

Sets the name JSPservlet statistic display will be called with (<servlet -mapping> tag in web.xml).
Default value: `ServletStat`.

2.2.21 Web.xml directory

Specifies the directory where JSPservlet will be installed.
Default value: `Cache path + File.separator + WEB-INF`.

This parameter is mandatory only for set command.

2.2.22 JSPservlet URL

Specifies the URL where JSPservlet will be downloaded from.
Default value: <http://pagebox.net/tomcat.jar>. It points the most recent version of JSPservlet, compiled for Tomcat 3. If your application server is Resin 1.1 or 1.2, change to <http://pagebox.net/resin.jar>.

This parameter is mandatory only for set command.

2.3 Commands

We recommend running the commands in the following order:

1. Once you have set fields (the most important ones being ID and Cache path), check
2. Once you have checked messages, generate to display the JSPservlet web.xml
3. set to install JSPservlet on the target machine

2.3.1 Check

Checks if parameters are valid. After setting the ID and the Cache path, you are typically displayed this:

JSPServletConfig

CRL certificate URL

CA user

CA password

CRL user

CRL password

Trace

Statistics

Resource expiration time

CRL scan period

Cache path *

Log file

Remote location property file

AllPermission policy file

Default policy file

ServletLog name

ServletStat name

The following two parameters are used only if you click set and want to upload JSPservlet.

web.xml directory *

JSPservlet URL *

Warning: C:\TEMP\myPageBox doesn't exist.
 Warning: directory where to create logs (myPageBox) doesn't exist
 Warning: directory where to create remote locations property file (myPageBox) doesn't exist
 Warning: allPermission policy C:\TEMP\myPageBox\allPermission.policy doesn't exist
 Warning: default policy C:\TEMP\myPageBox\default.policy doesn't exist

Alexis Grandemange (c) 2000-2001 GNU GPL 2

Figure 2: check

Messages are prefixed by:

- ?? Mandatory: a mandatory parameter was not set
- ?? Error: JSPservlet will not work with these parameters
- ?? Warning: JSPservlet can work with these parameters if you take the proposed action

2.3.2 Generate

Generate returns a web.xml file based on your JSPservletConfig form settings. For instance, with the parameters we set above:

```

<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
"http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">
<!-- Review on agrandem

        ACTION: Create CachePath:C:\TEMP\myPagebox Reason: doesn't exist

        ACTION: Create parent dir of LogFile:C:\TEMP\myPagebox\JSPservlet.log Reason:
doesn't exist

        ACTION: Create parent dir of
RemoteLocations:C:\TEMP\myPagebox\remoteLocation.properties Reason: doesn't exist

        ACTION: Create AllPermission policy file:C:\TEMP\myPagebox\allPermission.policy or
comment it. Reason: doesn't exist
        NOTE: if you comment it the archives will run without sandbox

        ACTION: Create default policy file:C:\TEMP\myPagebox\default.policy or comment it.
Reason: doesn't exist
        NOTE: if you comment it the archives will run without sandbox
-->

<web-app>
  <servlet>
    <servlet-name>JSPservlet</servlet-name>
    <servlet-class>JSPservletPkg.JSPservlet</servlet-class>

    <init-param>
      <param-name>cachePath</param-name>
      <param-value>C:/TEMP/myPagebox</param-value>
      <description>local cache</description>
    </init-param>

    <init-param>
      <param-name>toTrace</param-name>
      <param-value>TRUE</param-value>
      <description>trace activation</description>
    </init-param>

    <init-param>
      <param-name>toStat</param-name>
      <param-value>TRUE</param-value>
      <description>stat activation</description>
    </init-param>

    <init-param>
      <param-name>ID</param-name>
      <param-value>myPagebox</param-value>
      <description>unique identifier</description>
  </servlet>

```

```
<init-param>
<param-name>logfile</param-name>
<param-value>C:/TEMP/myPagebox/JSPservlet.log</param-value>
<description>trace location</description>
</init-param>

<init-param>
<param-name>remoteLocations</param-name>
<param-value>C:/TEMP/myPagebox/re moteLocation.properties</param-value>
<description>jar remote location</description>
</init-param>

<init-param>
<param-name>allPermissionPolicy</param-name>
<param-value>C:/TEMP/myPagebox/allPermission.policy</param-value>
<description>policy file granting all rights</description>
</init-param>

<init-param>
<param-name>defaultPolicy</param-name>
<param-value>C:/TEMP/myPagebox/default.policy</param-value>
<description>policy file for archives without explicit policy</description>
</init-param>

<init-param>
<param-name>keystore</param-name>
<param-value>keystore</param-value>
<description>keystore location + download policy</description>
</init-param>

<init-param>
<param-name>keystorePassword</param-name>
<param-value>keystorePassword</param-value>
<description>keystore password</description>
</init-param>

<!-- Uncomment the line if you fill the URL

<init-param>
<param-name>CRLURL</param-name>
<param-value>LDAP://directory.verisign.com</param-value>
<description>Certificate Revocation List LDAP URL</description>
</init-param>

-->

<!-- Uncomment the line if you fill the URL

<init-param>
<param-name>CAURL</param-name>
<param-value>LDAP://directory.verisign.com</param-value>
<description>CA Certificate LDAP URL</description>
</init-param>
```

```
<!-- Uncomment the line if you fill the identifier

<init-param>
<param-name>CRLLDAPuser</param-name>
<param-value></param-value>
<description>CRL LDAP user</description>
</init-param>

-->

<!-- Uncomment the line if you fill the password

<init-param>
<param-name>CRLLDAPpasswd</param-name>
<param-value></param-value>
<description>CRL LDAP password</description>
</init-param>

-->

<!-- Uncomment the line if you fill the identifier

<init-param>
<param-name>CALDAPuser</param-name>
<param-value></param-value>
<description>CA certificate LDAP user</description>
</init-param>

-->

<!-- Uncomment the line if you fill the password

<init-param>
<param-name>CALDAPpasswd</param-name>
<param-value></param-value>
<description>CA certificate LDAP password</description>
</init-param>

-->

<init-param>
<param-name>expiration</param-name>
<param-value>30</param-value>
<description>resource expiration time</description>
</init-param>

<init-param>
<param-name>CRLperiod</param-name>
<param-value>30</param-value>
<description>Certificate Revocation List scan period</description>
</init-param>

<init-param>
<param-name>ErrorPage</param-name>
<param-value>/JSPerror.html</param-value>
```

```
</init-param>

<init-param>
<param-name>NotFoundPage</param-name>
<param-value>/JSPnotFound.html</param-value>
<description>Page to display when target servlet not found</description>
</init-param>

<init-param>
<param-name>UnauthorizedPage</param-name>
<param-value>/JSPunauthorized.html</param-value>
<description>Target page tried an unauthorized access</descripti on>
</init-param>

<init-param>
<param-name>RevokedPage</param-name>
<param-value>/JSPrevoked.html</param-value>
<description>Revoked archive certificate</description>
</init-param>
</servlet>

<servlet>
  <servlet-name>ServletUpdate</servlet-name>
  <servlet-class>JSPservletPkg.ServletUpdate</servlet-class>
</servlet>

<servlet>
  <servlet-name>ServletLog</servlet-name>
  <servlet-class>JSPservletPkg.ServletLog</servlet-class>
</servlet>

<servlet>
  <servlet-name>ServletStat</servlet-name>
  <servlet-class>JSPservletPkg.ServletStat</servlet-class>
</servlet>

<servlet-mapping>
  <servlet-name>ServletUpdate</servlet-name>
  <url-pattern>/ServletUpdate</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>ServletLog</servlet-name>
  <url-pattern>/ServletLog</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>ServletStat</servlet-name>
  <url-pattern>/ServletStat</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>JSPservlet</servlet-name>
  <url-pattern>/</url-pattern>
</servlet-mapping>
```

```
</web-app>
```

The generated web.xml includes a comment section that lists the actions to take if you want to install JSPservlet by hand with this web.xml. Note set takes the actions listed above for you.

CAURL, CRLURL, CRLLDAPuser, CRLLDAPpasswd, CALDAPuser and CALLDAPpasswd are commented.

If you want to set CA and CRL parameters, we recommend coming back to JSPservletConfig form and run generate again.

2.3.3 Set

Set:

1. Creates needed directories
2. Creates allPermissionPolicy and defaultPolicy with default values
3. Downloads JSPservlet from JSPservletURL

If it returns no message, you only have to check (and possibly change) CA and CRL parameters.

2.4 Packaging

2.4.1 Source

JSPservletConfig.java

2.4.2 Class

JSPservletConfig.class

3 PublishConfig

3.1 Configuration

PublishConfig has no initialization parameter. It can be configured in web.xml as below:

```
<servlet>
  <servlet -name>PublishConfig</servlet -name>
  <servlet -class>configurator.PublishConfig</servlet -class>
</servlet>

<servlet-mapping>
  <servlet -name>PublishConfig</servlet -name>
  <url -pattern>/PublishConfig</url -pattern>
</servlet-mapping>
```

With Tomcat 3.1 or 3.2, it can simply be copied in the servlet directory and invoked through org.apache.tomcat.servlets.InvokerServlet with a URL <http://myhostport/servlet/PublishConfig>.

3.2 Display

When you call PublishConfig, the first time, you are displayed the form below with default values:

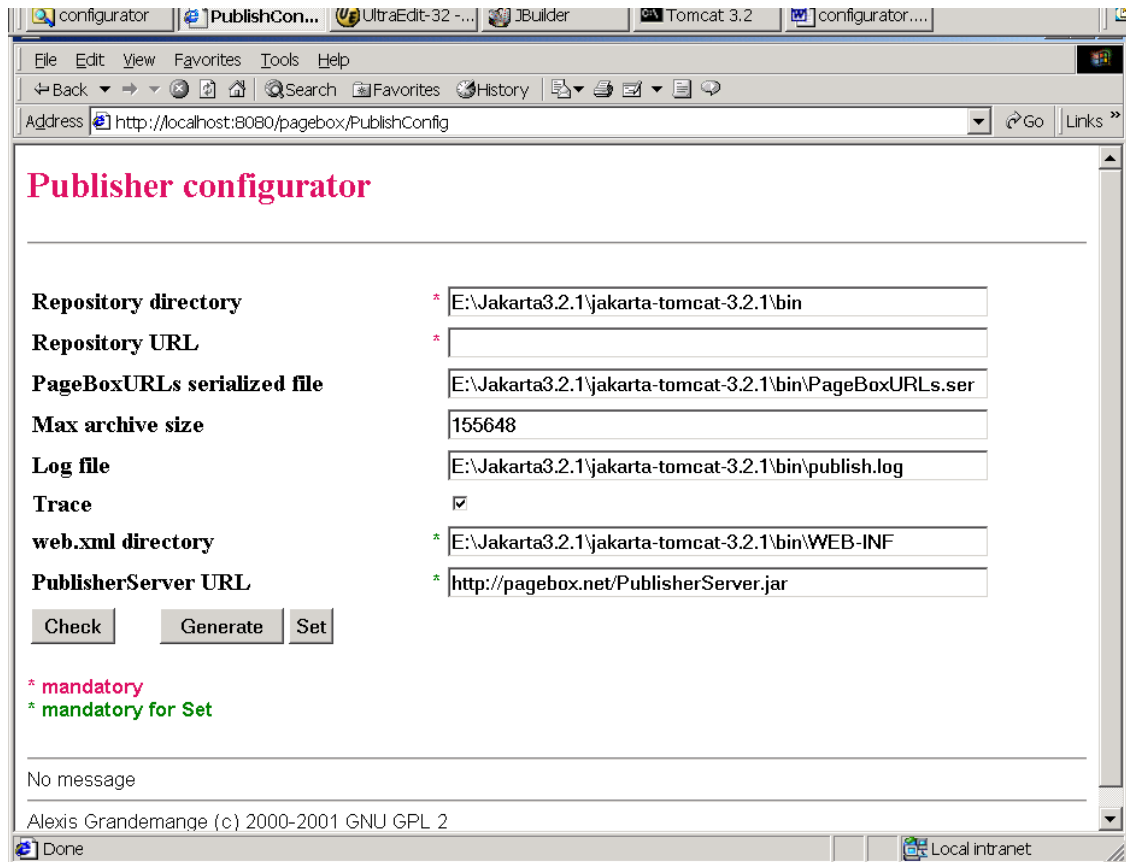


Figure 3: PublishConfig

3.2.1 Repository directory

Repository is the path of the directory where PublisherServer writes archives.

Web.xml initialization parameter for PublisherServer: Repository.

Default value: current path in PublishConfig context.

This default is a clue, not an appropriate value. It has to be the directory accessed with Repository URL.

3.2.2 Repository URL

URL where JSPservletPkg can retrieve the archive.

Web.xml initialization parameter for PublisherServer: repositoryURL.

This parameter is mandatory.

3.2.3 PageBoxURLs serialized file

PageBoxURLs is a file containing the serialized state of archive publications and subscriptions.

Web.xml initialization parameter for PublisherServer: PageBoxURLs.

Default value: Repository URL + File.separator + PageBoxURLs.ser.

If you set Repository URL to a value different of current path and click on Check, then PageBoxURLs serialized file is updated.

3.2.4 Max archive size

If the archive is bigger, PublisherServer rejects the upload request.

Web.xml initialization parameter for PublisherServer: Maxlength.
Default value: 155648.

3.2.5 Log file

Path of the file where PublisherServer logs.

Web.xml initialization parameter for PublisherServer: LogFile.
Default value: Repository URL + File.separator + publish.log.

If you set Repository URL to a value different of current path and click on Check, then Log file is updated.

3.2.6 Trace

If set to true or TRUE, PublisherServer logs intensively.

Web.xml initialization parameter for PublisherServer: LogFile.
Default value: TRUE.

3.2.7 Web.xml directory

Specifies the directory where PublisherServer will be installed.
Default value: Repository URL + File.separator + WEB -INF.

This parameter is mandatory only for set command.

3.2.8 PublisherServer directory

Specifies the URL where JSPservlet will be downloaded from.
Default value: <http://pagebox.net/PublisherServer.jar>. It points the most recent version of PublisherServer.

This parameter is mandatory only for set command.

3.3 **Commands**

We recommend running the commands in the following order:

1. Once you have set fields (the most important ones being Repository File and Repository URL), check
2. Once you have checked messages, generate to display the PublisherServer web.xml
3. set to install PublisherServer on the target machine

3.3.1 Check

Checks if parameters are valid. After setting fields, you are typically displayed this:

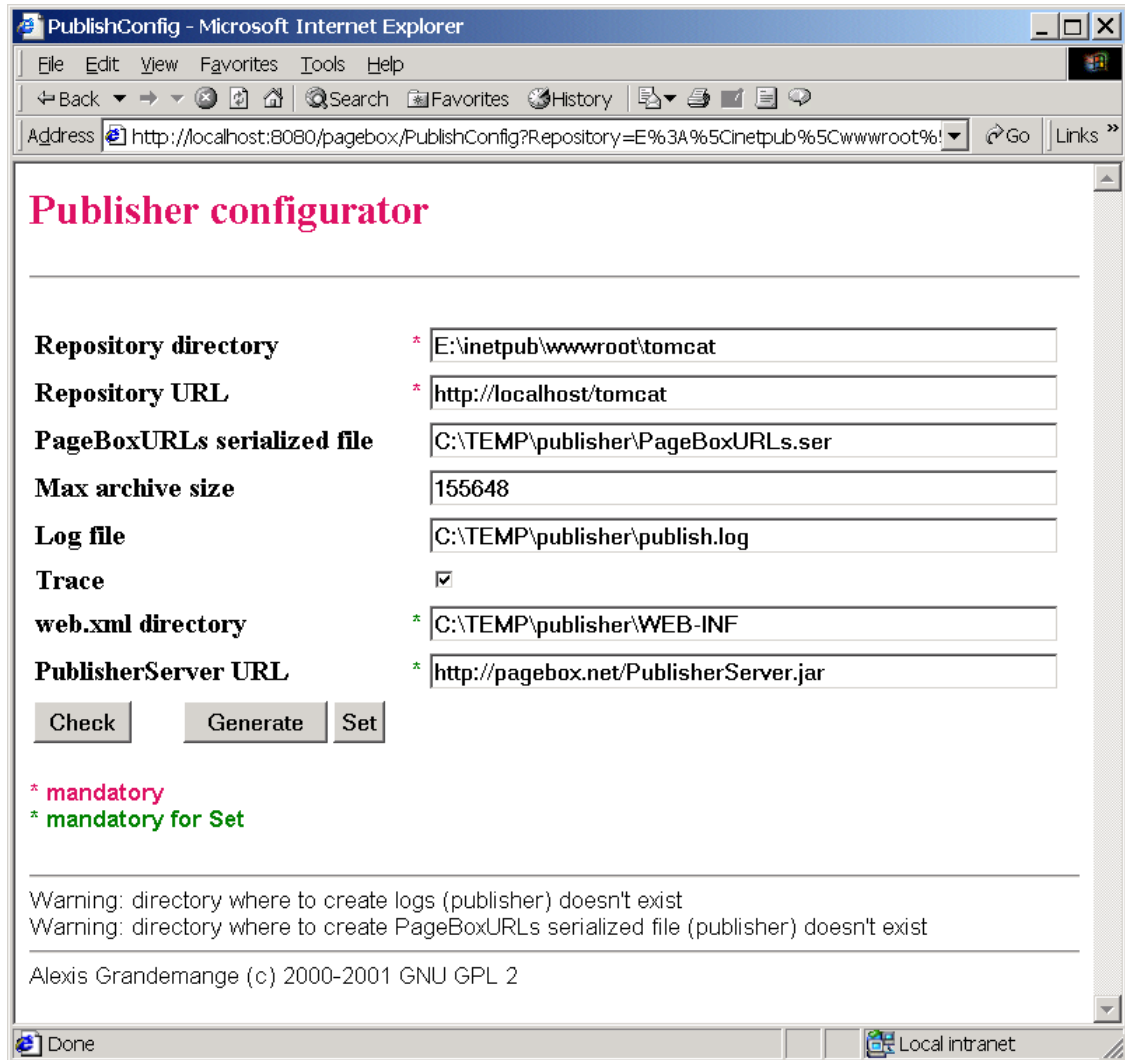


Figure 4: PublishConfig check

Messages are prefixed by:

- ?? Mandatory: a mandatory parameter was not set
- ?? Error: PublisherServer will not work with these parameters
- ?? Warning: PublisherServer can work with these parameters if you take the proposed action

3.3.2 Generate

Generate returns a web.xml file based on your PublishConfig form settings. For instance, with the parameters we set above:

```
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
"http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">
<!-- Review on agrandem
```

ACTION: Create log directory:C:\TEMP\publisher

```
-->

<web-app>
  <servlet>
    <servlet-name>publish</servlet-name>
    <servlet-class>PublisherServer.PublisherServlet</servlet -class>

    <init-param>
      <param-name>Repository</param-name>
      <param-value>E:/inetpub/wwwroot/tomcat</param -value>
      <description>Archive repository</description>
    </init-param>

    <init-param>
      <param-name>Maxlength</param-name>
      <param-value>155648</param -value>
      <description>max archive size</description>
    </init-param>

    <init-param>
      <param-name>toTrace</param-name>
      <param-value>TRUE</param -value>
      <description>trace activation</description>
    </init-param>

    <init-param>
      <param-name>Logfile</param-name>
      <param-value>C:/TEMP/publisher/publish.log</param -value>
      <description>path of log file</description>
    </init-param>

    <init-param>
      <param-name>PageBoxURLs</param-name>
      <param-value>C:/TEMP/publisher/PageBoxURLs.ser</param -value>
      <description>URLs where to publish archives serialized file</description>
    </init-param>

    <init-param>
      <param-name>repositoryURL</param-name>
      <param-value>localhost/tomcat</param -value>
      <description>URL where to download archives</description>
    </init-param>
  </servlet>

  <servlet>
    <servlet-name>publishLog</servlet -name>
    <servlet-class>PublisherServer.PublisherLog</servlet -class>
  </servlet>

  <servlet-mapping>
    <servlet-name>publish</servlet -name>
    <url-pattern>/publish</url -pattern>
  </servlet-mapping>
```

```
<servlet-mapping>
  <servlet-name>publishLog</servlet-name>
  <url-pattern>/publishLog</url-pattern>
</servlet-mapping>
</web-app>
```

The generated web.xml includes a comment section that lists the actions to take if you want to install PublisherServlet by hand with this web.xml. Note set takes the actions listed above for you.

3.3.3 Set

Set:

1. Creates needed directories
2. Downloads JSPservlet from JSPservletU RL

3.4 *Packaging*

3.4.1 Source

PublishConfig.java

3.4.2 Class

PublishConfig.class