

WebSphere Liberty Profile のデプロイメントのトラブルシューティング

Spring Boot アプリケーションにアクセスしようとして `http://localhost:9080/myapp/` で「コンテキストルートが見つかりません」というエラーが発生した場合、WebSphere Liberty Profile (WLP) が期待されるコンテキストルート (`/myapp`) の下にアプリケーションを正しくデプロイしていないことを示しています。この問題は、デプロイメントの設定、サーバーの設定、または WAR ファイル自体に問題があることが原因です。以下に、この問題をトラブルシューティングし解決するための包括的なガイドを示します。

問題の理解

「コンテキストルートが見つかりません」というエラーは、WLP が `/myapp` にデプロイされたアプリケーションを見つけられないことを意味します。通常、WAR ファイル `myapp.war` を WLP の `dropins` ディレクトリに置くと、自動的にコンテキストルート `/myapp` でデプロイされ、`http://localhost:9080/myapp/` でアクセス可能になります。しかし、これが起こらないため、デプロイが失敗した理由を調べる必要があります。

トラブルシューティング手順

1. サーバーログでデプロイメントメッセージを確認する 最初のステップは、WLP がアプリケーションをデプロイしたかどうかを確認することです。

- **ログの場所:**

- サーバー名が `myServer` の場合、ログは以下の場所にあります:

`/opt/ibm/wlp/usr/servers/myServer/logs/messages.log`

または

`/opt/ibm/wlp/usr/servers/myServer/logs/console.log`

- デフォルトサーバーを使用している場合、`myServer` を `defaultServer` に置き換えます。

- **デプロイメントの確認:**

- 次のようなメッセージが表示されるはずです:

```
[AUDIT    ] CWWKT0016I: Web application available (default_host): http://localhost:9080/myapp/
```

これは、アプリケーションがデプロイされ、利用可能であることを示します。

- さらに、以下のメッセージも確認します:

```
CWWKZ0001I: Application myapp started in X.XXX seconds.
```

これは、アプリケーションが正常に開始されたことを確認します。

・対応:

- これらのメッセージが表示されない場合、アプリケーションはデプロイされていません。ログに ERROR または WARNING メッセージが表示されているか確認し、なぜデプロイに失敗したかを特定します（例：欠落している機能、ファイルのパーミッション、起動の失敗）。
- Spring Boot の起動ログ（例：Spring Boot のバナー）が表示されている場合、アプリケーションは読み込まれており、問題はコンテキストルートまたは URL マッピングにある可能性があります。

2. WAR ファイルの場所とパーミッションを確認する

WAR ファイルが dropins ディレクトリに正しく置かれ、WLP によってアクセス可能であることを確認します。

・パスの確認:

- サーバー名が myServer の場合、WAR ファイルは以下の場所にあるはずです:

```
/opt/ibm/wlp/usr/servers/myServer/dropins/myapp.war
```

- defaultServer を使用している場合、以下のように調整します:

```
/opt/ibm/wlp/usr/servers/defaultServer/dropins/myapp.war
```

・パーミッションの確認:

- WLP プロセスがファイルを読み取れるようにするために、Unix 系システムで以下を実行します:

```
ls -l /opt/ibm/wlp/usr/servers/myServer/dropins/myapp.war
```

ファイルは WLP を実行するユーザー（例：rw-r--r--）によって読み取れる必要があります。

・対応:

- ファイルが欠落しているか、場所が間違っている場合、正しい dropins ディレクトリにコピーします:

```
cp target/myapp.war /opt/ibm/wlp/usr/servers/myServer/dropins/
```

- 必要に応じてパーミッションを修正します:

```
chmod 644 /opt/ibm/wlp/usr/servers/myServer/dropins/myapp.war
```

3. server.xml で dropins の監視を確認する

WLP の dropins ディレクトリはデフォルトで有効になっていますが、カスタム設定によって無効になっていることがあります。

・server.xml の確認:

- サーバーの設定ファイルを開きます:

```
/opt/ibm/wlp/usr/servers/myServer/server.xml
```

- applicationMonitor 要素を確認します:

```
<applicationMonitor updateTrigger="polled" pollingRate="5s" dropins="dropins" />
```

これは、WLP が dropins ディレクトリを 5 秒ごとに新しいアプリケーションを監視していることを確認します。

- **対応:**

- 存在しない場合、<server> タグ内に上記の行を追加するか、dropins を無効にするオーバーライド設定がないことを確認します。

- 変更後、サーバーを再起動します:

```
/opt/ibm/wlp/bin/server stop myServer  
/opt/ibm/wlp/bin/server start myServer
```

4. 必要な機能が有効になっていることを確認する

WLP は、Spring Boot WAR ファイルをデプロイするために、Servlet サポートなどの特定の機能が必要です。

- **server.xml の確認:**

- featureManager セクションに以下が含まれていることを確認します:

```
<featureManager>  
    <feature>javaee-8.0</feature>  
</featureManager>
```

javaee-8.0 機能には Servlet 4.0 が含まれており、Spring Boot と互換性があります。代替として、少なくとも servlet-4.0 が存在している必要があります。

- **対応:**

- 存在しない場合、機能を追加し、サーバーを再起動します。

5. WAR ファイルの構造を検証する

破損したり、不適切な構造の WAR ファイルはデプロイを防ぐことがあります。

- **WAR の検証:**

- WAR ファイルを展開して内容を確認します:

```
unzip -l myapp.war
```

- 次のような内容を確認します:

- * WEB-INF/classes/com/example/demo/HelloController.class (コントローラークラス)。
- * WEB-INF/lib/ に Spring Boot の依存関係 (例: spring-web-x.x.x.jar) が含まれている。

- **対応:**

- 構造が不正な場合、WAR を再構築します:

```
mvn clean package
```

pom.xml で <packaging>war</packaging> を設定し、spring-boot-starter-tomcat を <scope>provided</scope> にマークします。

6. apps ディレクトリを使用した代替デプロイ dropins が失敗した場合、apps ディレクトリを使用してアプリケーションを明示的にデプロイしてみてください。

・手順:

- WAR ファイルを移動します:

```
mv /opt/ibm/wlp/usr/servers/myServer/dropins/myapp.war /opt/ibm/wlp/usr/servers/myServer/apps/
```

- server.xml を編集して追加します:

```
<application id="myapp" name="myapp" type="war" location="${server.config.dir}/apps/myapp.war">
    <context-root>/myapp</context-root>
</application>
```

- サーバーを再起動します:

```
/opt/ibm/wlp/bin/server restart myServer
```

・再度テスト:

- <http://localhost:9080/myapp/> にアクセスします。動作する場合、dropins に問題があったことを示します。

7. サーバーの状態を確認する サーバーが正しく動作していることを確認します。

・状態の確認:

```
/opt/ibm/wlp/bin/server status myServer
```

- 停止している場合、起動します:

```
/opt/ibm/wlp/bin/server start myServer
```

・WLP 自体をテスト:

- <http://localhost:9080/> にアクセスします。WLP が動作している場合、ウェルカムページが表示されます（オーバーライドされていない限り）。この操作が失敗する場合、より広範なサーバーの問題があります。

解決策

「コンテキストルートが見つかりません」というエラーを修正するには:

1. ここから始める:

- ・サーバーログ (messages.log または console.log) でデプロイメントメッセージやエラーを確認します。

- myapp.war が dropins ディレクトリにあり、適切なパーミッションが設定されていることを確認します。

2. 次のステップ:

- server.xml で javaee-8.0 機能が有効になっているか、dropins を監視しているかを確認します。
- 解決しない場合、apps ディレクトリを使用して server.xml に明示的な <application> 構成を追加してデプロイします。

3. 最終確認:

- 変更後、WLP を再起動し、`http://localhost:9080/myapp/` にアクセスします。デプロイが成功すると「Hello World!」が表示されます。

これらの手順に従うことで、問題がサーバーの設定、WAR の配置、または WAR ファイル自体にあるかを特定し、解決することができます。