

i-JADE



i-JADE の提供するエミュレーター上で SSL 接続を可能にする方法



Java Application Development Environment for i-mode





1. このソフトウェアの著作権は、株式会社ゼンテック・テクノロジー・ジャパンにあります。
 2. このソフトウェアおよびマニュアルの一部または全部を無断で使用、複製することはできません。
 3. このソフトウェアおよびマニュアルは、本製品の使用許諾契約書のもとでのみ使用することができます。
 4. このソフトウェアおよびマニュアルを運用した結果の影響については、いっさい責任を負いかねますのでご了承ください。
 5. このソフトウェアの仕様、およびマニュアルに記載されている事柄は、将来予告なしに変更する可能性があります。
-
- i-JADE および i-JADE ロゴは、株式会社ゼンテック・テクノロジー・ジャパンの米国およびその他の国における登録商標または商標です。
 - iモードは、株式会社 NTT ドコモの商標および登録商標です。
 - Java、Java Development Kit、JDK は、米国およびその他の国における米国 Sun Microsystems, Inc. の商標または登録商標です。
 - Windows は、米国 Microsoft Corporation の米国およびその他の国における商標または登録商標です。
 - その他、記載されている会社名、製品名は、各社の商標および登録商標です。

Copyright (C) 2000-2001 Zentek Technology Japan, Inc. All Rights Reserved.

Copyright (C) 2000-2001 Zentek Technology , Inc. All Rights Reserved.



目次



第 1 章 はじめに

第 2 章 設定

JSSE をダウンロード	9
環境設定	9
起動パラメータ	10
Java 実行環境に含まれない証明書を使用する場合	10
keytool コマンド利用例	11
i-JADE の起動	11

第 3 章 JBuilder への組み込み

必須ライブラリの追加	13
VM の引数の設定	14
java.security ファイルの設定	15
実行例	15

第 4 章 VisualCafe への組み込み

必須ライブラリの追加	17
VM の引数の設定	18
java.security ファイルの設定	19
実行例	19

第 5 章 Apworks への組み込み

必須ライブラリの追加	21
VM の引数の設定	22
java.security ファイルの設定	23
実行例	23



第 1 章 はじめに

iモードにおいては通信手段として SSL が提供されております。

ここでは、i-JADE の提供するエミュレーター上で SSL 接続を可能にする方法をご紹介します。

なお、設定の変更による、データおよびハードウェアなどの損失、破損、不具合について、弊社は一切の責任をおいませぬ。各自の責任において設定してください。



第 2 章 設定

i-JADE のパッケージには SSL 通信は含まれません。ここでは、Sun Microsystems 社が提供している” JSSE (Java Secure Socket Extension)” (<http://java.sun.com/products/jsse>) を使用した SSL 通信をご紹介させていただきます。

2.1 JSSE をダウンロード

<http://java.sun.com/products/jsse> より、注意事項を良くお読みになった上で「jsse-1.0.2-g1.zip (2001 年 6 月 1 日現在)」をダウンロードしてください。

2.2 環境設定

ZIP ファイルには、jsse.jar、jnet.jar、jcert.jar が含まれています。これらに classpath を設定します。また Java の実行環境 (JDK インストール先) の [lib/security] ディレクトリ配下の java.security ファイルの以下の記述を加えます。

`security.provider.3=com.sun.net.internal.ssl.Provider`



2.3 起動パラメータ

実際に i-JADE を起動する際には、以下の VM のパラメータを渡す必要性があります。

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

また、プロキシ経由でアクセスする場合には以下のパラメータも渡す必要性があります。

`-Dhttps.proxyHost=[プロキシホスト名] -Dhttps.proxyPort=[ポート番号]`
(通常は 8080)

上記の手順を踏むことにより、Verisign などの大手の証明書業者は Java の実行環境に組み込まれているので、そのような証明書のサイトとの HTTPS 通信は可能になります。

2.4 Java 実行環境に含まれない証明書を使用する場合

Java 実行環境に含まれない証明書（自己証明書等）を使用する場合には以下のパラメータを加えます。

`-Djavax.net.ssl.trustStore=[キーストアファイル名]`

このキーストアファイルは、keytool コマンドを用いて作成します。keytool コマンドは既存の JDK に含まれています。

`keytool -keystore [キーストアファイル名] -alias [ユニークな名前] -import -file [証明書ファイル]`

注意) apache-ssl を使用している場合には証明書ファイルは conf ファイルで設定した SSLCertificateFile で指定したファイルになります。

また、上記コマンドを発行した際にはパスワード入力を促されますので、証明書発行時に設定したパスワードを入力してください。



2.5 keytool コマンド利用例

[キーストアへの追加]

```
keytool -keystore [ キーストアファイル名 ] -alias [ ユニークな名前 ] -import  
-file [ 証明書ファイル ]
```

[キーストア内容確認]

```
keytool -keystore [ キーストアファイル名 ] -list
```

[キーストアのエントリ削除]

```
keytool -keystore [ キーストアファイル名 ] -delete -alias [ ユニークな名前 ]
```

2.6 i-JADE の起動

実際に i-JADE を起動する際には以下のコマンドを実行します。

大手証明書業者を証明書を使用する場合

```
java -Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol -cp  
c:\jdk1.3.0_02\lib\ext\jcert.jar;c:\jdk1.3.0_02\lib\ext\jsse.jar;c:\jdk1.3.0_02\lib\ext\j  
net.jar;c:\i-jade\i-jade-n.jar com.zentek.jam.Jam [Appclass 名、もしくは Jam  
ファイル名 (フルパス指定)]
```

事項証明書を使用する場合

```
java -Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol  
-Djavax.net.ssl.trustStore=[ キーストアファイル名 (フルパスで指定) ] -cp  
c:\jdk1.3.0_02\lib\ext\jcert.jar;c:\jdk1.3.0_02\lib\ext\jsse.jar;c:\jdk1.3.0_02\lib\ext\j  
net.jar;c:\i-jade\i-jade-n.jar com.zentek.jam.Jam [Appclass 名、もしくは Jam  
ファイル名 (フルパス指定)]
```



第 3 章 JBuilder への組み込み

ここでは、i-JADE Lite においての IDE への SSL の組み込み方法に関して、Booland 社の JBuilder 4 Professional 版を説明をいたします。また、i-JADE Lite の JBuilder への詳しい組み込み方法は弊社ホームページ (<http://zentek.com/i-JADE/>) から別途 [i-JADE Lite と各種 IDE との連携に関するドキュメント マニュアル Borland JBuilder4 Professional 編] をダウンロードしてください。

3.1 必須ライブラリの追加

[プロジェクト] → [プロジェクトプロパティ] を選択し、[必須ライブラリ] タグを選択します。ここで、[新規] ボタンを押下し、[jees.jar、jnet.jar、jcert.jar] を必須ライブラリとして追加いたします。





3.2 VM の引数の設定

[プロジェクト] → [プロジェクトプロパティ] を選択し、[実行] タグを選択します。その VM 引数に以下を追加します。

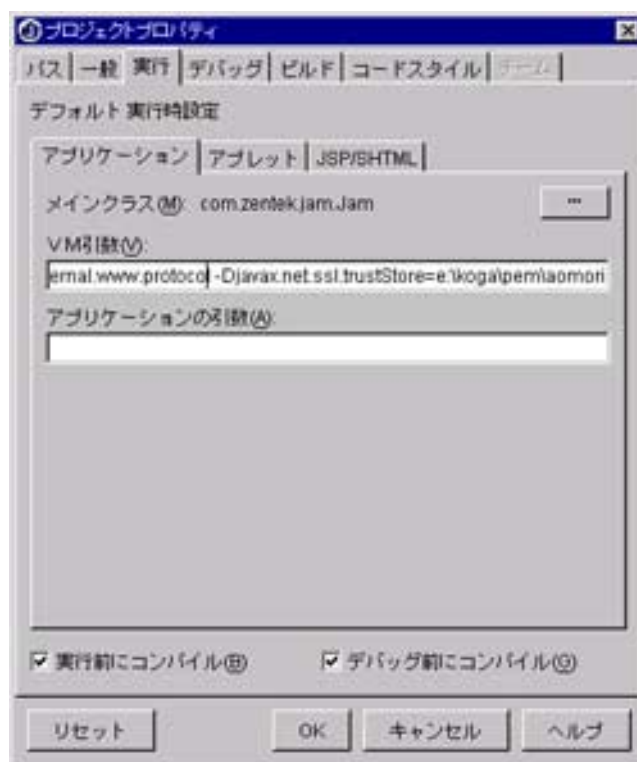
大手証明書業者の証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

自己証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

`-Djavax.net.ssl.trustStore=[キーストアファイル名(フルパス指定)]`





3.3 java.security ファイルの設定

JBuilder4 の JDK1.3 環境の java.security ファイル
(...\JBuilder4\jdk1.3\lib\security\java.security) 中に以下 1 行を追加します。

`security.provider.3=com.sun.net.ssl.internal.ssl.Provider`

3.4 実行例

以下は、SSL を使用したログイン認証画面になります。





第 4 章 VisualCafe への組み込み

ここでは、i-JADE Lite における IDE への SSL の組み込み方法に関して、webGain 社の VisualCafe Expert Edition 版を説明いたします。また、i-JADE Lite の Visual Cafe への詳しい組み込み方法は弊社ホームページ (<http://zentek.com/i-JADE/>) から別途 [i-JADE Lite と各種 IDE との連携に関するドキュメント マニュアル WebGain VisualCafe Expert Edition 編] をダウンロードしてください。

4.1 必須ライブラリの追加

[プロジェクト] → [設定] を選択し、[ディレクトリ] タブを選択します。
[新規追加] ボタンを押下し、[jees.jar、jnet.jar、jcert.jar] を必須ライブラリとして追加いたします。





4.2 VM の引数の設定

[ツール] → [環境設定] を選択し、[仮想マシン] タグを選択します。その VM のコマンドオプションに以下を追加します。

大手証明書業者の証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

自己証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

`-Djavax.net.ssl.trustStore=[キーストアファイル名 (フルパス指定)]`





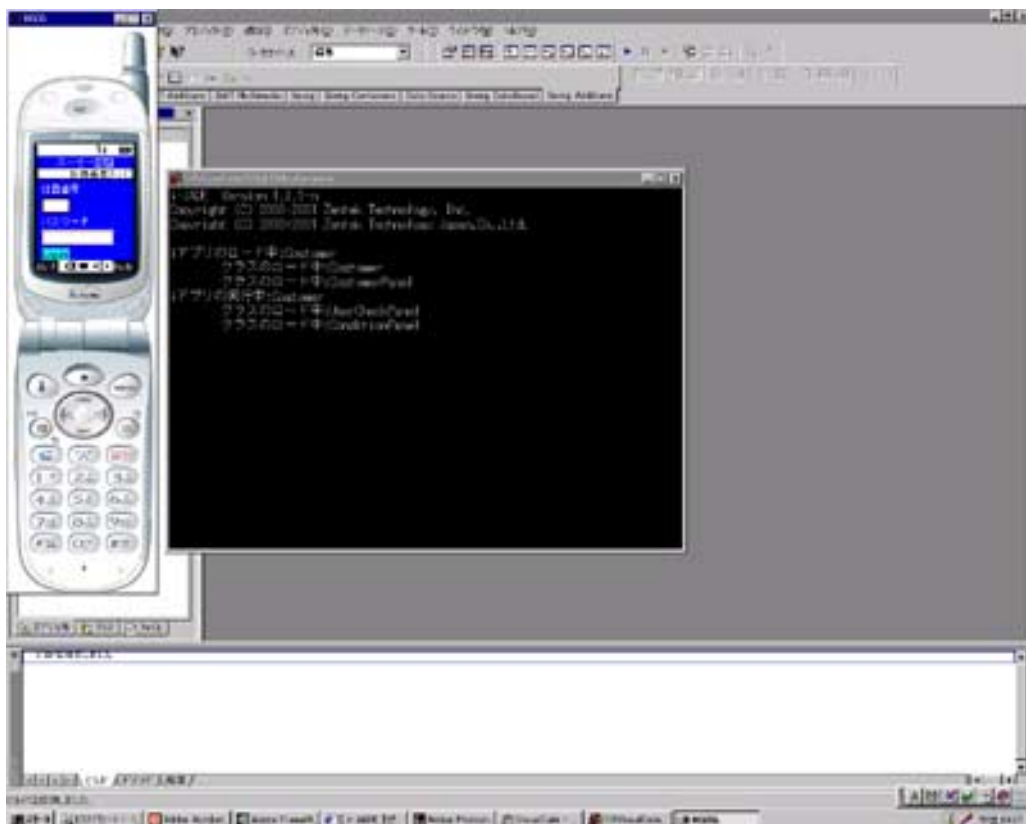
4.3 java.security ファイルの設定

VisualCafe の JDK1.3 環境の java.security ファイル
(...\JBuilder4\jdk1.3\jre\lib\security\java.security) 中に以下 1 行を追加しま
す。

`security.provider.3=com.sun.net.ssl.internal.ssl.Provider`

4.4 実行例

以下は、SSL を使用したログイン認証画面になります。



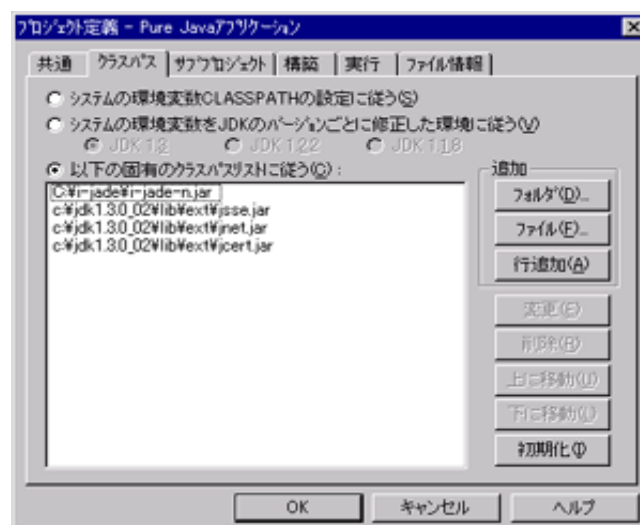


第 5 章 Apworks への組み込み

ここでは、i-JADE Lite においての IDE への SSL の組み込み方法に関して、富士通株式会社の Interstage Apworks 版を説明をいたします。また、i-JADE Lite の Apworks への詳しい組み込み方法は弊社ホームページ (<http://zentek.com/i-JADE/>) から別途 [i-JADE Lite と各種 IDE との連携に関するドキュメント マニュアル 富士通株式会社 INTERSTAGE APWORKS 編] をダウンロードしてください。

5.1 必須ライブラリの追加

[ファイル] → [プロパティ] を選択し、[クラスパス] タブを選択します。
[ファイル] ボタンを押下し、[jees.jar、jnet.jar、jcert.jar] を必須ライブラリとして追加いたします。





5.2 VM の引数の設定

[ファイル]→[プロパティ]を選択し、[実行]タブを選択します。そのオプションに以下を追加します。

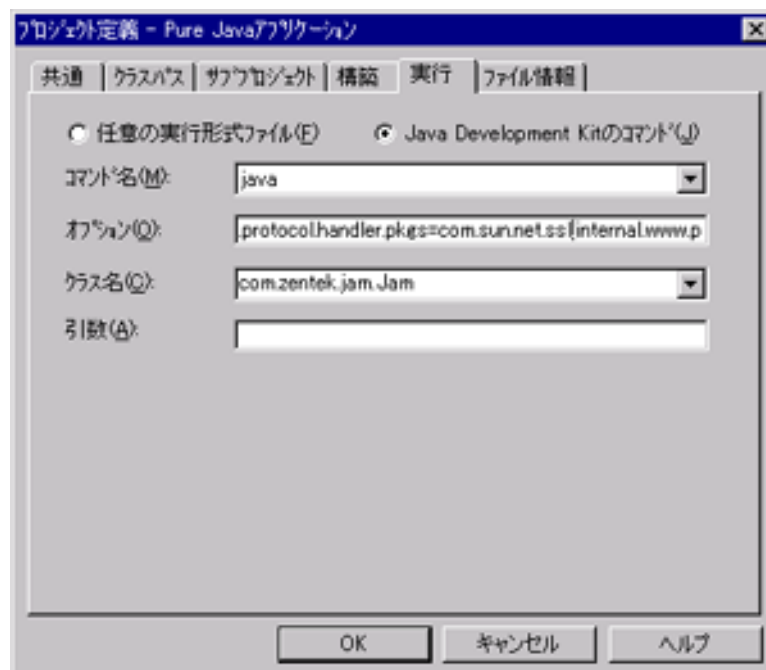
大手証明書業者の証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

自己証明書を使用する場合

`-Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol`

`-Djavax.net.ssl.trustStore=[キーストアファイル名(フルパス指定)]`





5.3 java.security ファイルの設定

Apworks の JDK1.3 環境の java.security ファイル
(...\APW\jbc3\jdk\jre\lib\security\java.security) 中に以下 1 行を追加します。

`security.provider.3=com.sun.net.ssl.internal.ssl.Provider`

5.4 実行例

以下は、SSL を使用したログイン認証画面になります。

