

IC3000でアプリケーション/コンテナを実行する コンソールにアクセスする方法

内容

[概要](#)

[IC3000でアプリケーション/コンテナを実行するコンソールにアクセスする方法](#)

[IC3000を\[Developer Mode\]に設定します](#)

[IOxアプリケーションの導入、アクティブ化、および開始](#)

概要

このドキュメントでは、トラブルシューティングとデバッグを行うために、IC3000上でアプリケーション/コンテナを実行しているコンソールにアクセスする方法について説明します。

IC3000でアプリケーション/コンテナを実行するコンソールにアクセスする方法

IC3000上のコンソールにアクセスするには、IC3000が開発者モードで実行されている必要があります。IC3000が実稼働モードになると、アプリケーションコンソールにアクセスできません。これは意図的なものだ。

IC3000を[Developer Mode]に設定します

実行中（または失敗した）アプリケーションをデバッグ/トラブルシューティングするには、まずIC3000が開発者モードであることを確認します。

開発者モードでは、IC3000は管理対象外の開発デバイスであり、Local Managerとioxclientを介して制御されます。

1. USBまたはシリアルコンソールケーブルを使用してコンソールに接続します。
2. 「developer」ユーザのパスワードを設定します（コマンドdeveloper set-passwordを使用）。

```
ic3k>developer set-password
Enter password:
Re-enter password:
Password set successfully!
```

3. IC3000の管理インターフェイスをネットワークケーブルでコンピュータに接続します。
4. コンピュータのネットワークインターフェイスに「169.254.128.4 (ネットマスク 255.255.0.0)」のIPアドレスを割り当てます。この特定のIPv4リンクローカルアドレスを割り当てることが重要です。
5. IC3000は30秒で開発者モードで動作する準備が整います(30秒の遅延は、IC3000が初めて起動された時点でのみ発生します。それ以降のリロードでは、すぐにIC3000が開発者モードに移行します（遅延はありません）。

6. 次のコマンドを使用して、IC3000が開発者モードであるかどうかを確認できます。

```
ic3k>show ida status
Status: Running
Operation Mode: Developer
FND Host: N/A
FND Connection Status: Not Connected
Periodic Metrics Interval: N/A
Heartbeat Interval: N/A
Is Registered: False
HTTP Server Status: Running
```

7. コンピュータのブラウザから「<https://169.254.128.2:8443>」にアクセスします。

8. ステップ2で設定した「developer」ユーザとパスワードを使用してログインします。

IOxアプリケーションの導入、アクティブ化、および開始

IOxアプリケーションパッケージの展開、アクティブ化、および起動には、ブラウザからioxclientまたはLocal Managerを使用します。

オプションで、アプリケーションをアクティブ化するときにはデバッグオプションを有効にできます。これにより、図に示すように、アプリケーションが予期せず終了したときに、アプリケーションコンテナが停止することを防止できます。

The screenshot shows a configuration page for IOx applications. It features a 'Resources' section with a 'Resource Profile' sub-section. The profile is set to 'c1.small'. Below this, there are three rows for resource allocation: CPU (200 units), Memory (64 MB), and Disk (10 MB). At the bottom, there are two buttons: 'Activate' (with a checkmark) and 'debug mode (For troubleshooting only)' (with a checked checkbox).

Resource	Value	Unit
Profile	c1.small	
CPU	200	cpu-units
Memory	64	MB
Disk	10	MB

Avail. CPU (cpu-units) 10260 Avail. Memory (Mb) 6400

Activate debug mode (For troubleshooting only)

アプリケーションコンソールに接続します。

アプリケーションが実行された後（またはデバッグモードのままになっている場合）、次のようにコンテナのコンソールに接続できます。

まず、Local Managerまたはioxclientを使用して、コンテナの秘密キーをダウンロードします。

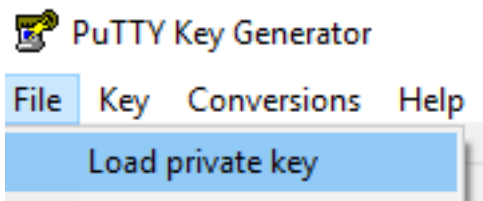
Local Manager > Applications > <specific app> > Manage > App Info > *.pem fileを選択してください。

App Access	
Console Access	ssh -p {SSH_PORT} -i test.pem appconsole@169.254.128.2

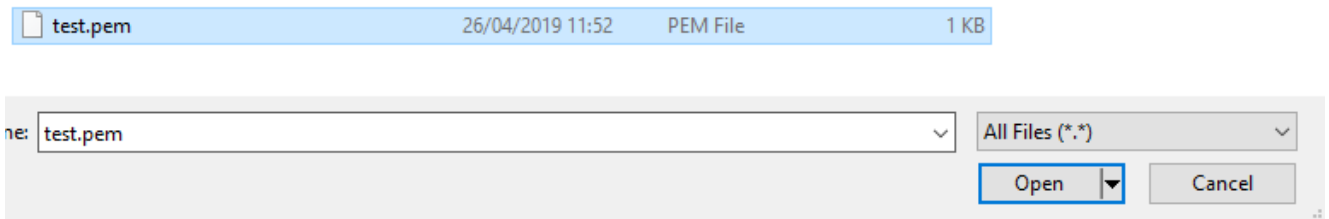
Linuxでは、次のようにSSHポート22を使用してリストされているコマンドを起動します。ssh -i test.pem appconsole@169.254.128.2

Windowsの場合、Puttyを使用する場合は、まずPuTygenを使用して.pemファイルをPutty-compatible.ppkに変換します。

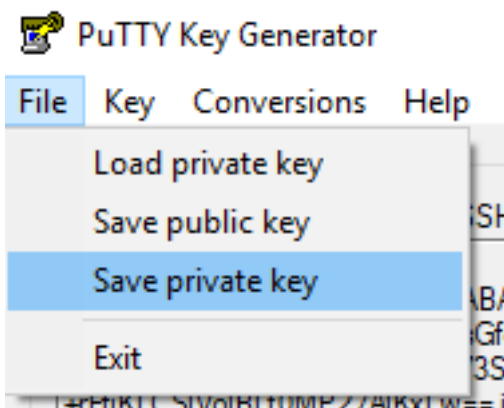
1. PuGenを起動します。
2. 図に示すように、[File] > [Load private key]に移動します。



3. ファイルフィルタをAll Filesに設定し、図に示すようにダウンロード.pemを開きます。

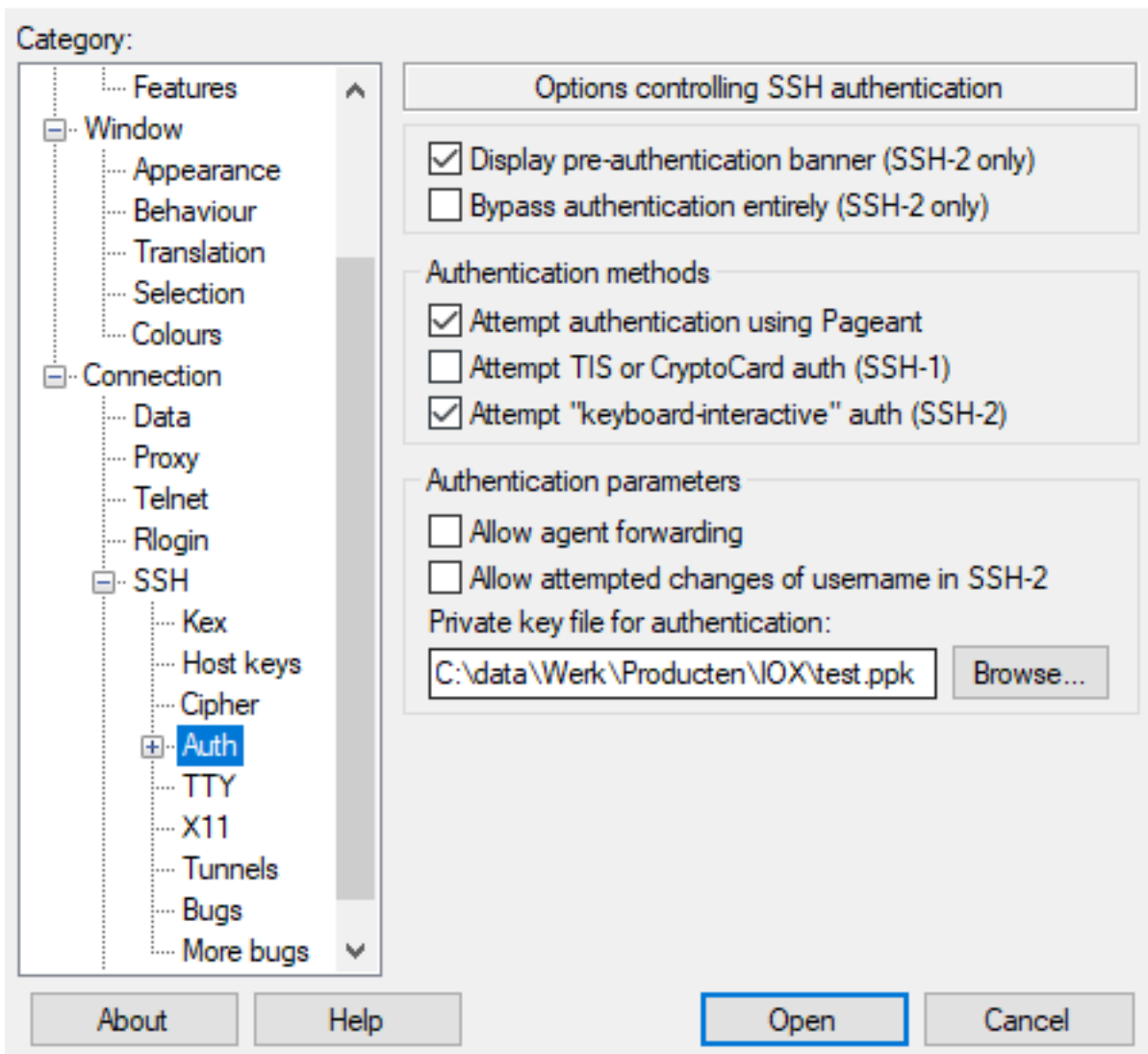


4. [File] > [Save private key]に移動し、図に示すように.pemを.ppkとして保存します。



.ppkが表示されたら、PuTTYを起動し、セッションダイアログで169.254.128.2ポート22と入力します。

次に、[Connection - SSH - Auth]に移動し、図に示すように.ppkファイルを適用します。



[Open]をクリックし、セッションを開始します。ユーザ名として、図に示すように「**appconsole**」と入力します。

The screenshot shows a PuTTY terminal window titled '169.254.128.2 - PuTTY'. The terminal output is as follows:

```
login as: appconsole
Authenticating with public key "imported-openssh-key"
/ # ps
PID  USER    TIME    COMMAND
  1  root      0:00    {startcontainer.} /bin/sh /.iox/startcontainer.sh
 36  root      0:00    python /webserver.py 9000
 37  root      0:00    /bin/sh
 38  root      0:00    ps
/ #
```

これで、IC3000上で実行中のIOxコンテナのアプリケーションコンソールが表示されます。