

**WebSphere Application Server**

**トラブルシューティングおよびパフォーマンス　ワークショップ（Docker版）準備手順**

**著者**

* ケビン グリゴレンコ <Kevin Grigorenko> ([kevin.grigorenko@us.ibm.com](mailto:kevin.grigorenko@us.ibm.com))

**目次**

[1 ワークショップの準備 2](#_Toc9032639)

[2 補足 9](#_Toc9032640)

[2.1 Windows のリモートデスクトップ接続でDockerコンテナにアクセスする 9](#_Toc9032641)

# ワークショップの準備

このワークショップは、Docker環境で行うことを前提としています。WindowsやMacをお使いの場合はDocker Desktopをインストールしてください。Linuxをお使いの場合はDockerをインストールしてください。

ワークショップでは、Traditional WAS とWAS Libertyのトラブルシューティングとパフォーマンス・チューニングの概念および解析に必要なツールの使い方を学びます。

トラブルシューティングとパフォーマンス・チューニングは、多くの場合、オペレーティングシステムやJavaのレベルで行われるため、このワークショップで取得する技術はWASだけでなく広い範囲で応用できます。

このワークショップのDockerイメージには、Traditional WAS と WAS Libertyがあらかじめインストール、構成されているので、課題にすぐにとりかかれるようになっています。

注）ワークショップのDockerイメージには、ワークショップ用に多くのサービスやツールがインストールされています。本番環境で使われるDockerには、必要なサービスのみインストールするよう推奨されています。

1. Docker Desktopをインストールしてください。
2. Mac ("Apple Mac OS Sierra 10.12"以降のバージョン)
   * + ダウンロード: <https://hub.docker.com/editions/community/docker-ce-desktop-mac>
     + 詳細は次のリンクをご参照ください。 <https://docs.docker.com/docker-for-mac/install/>
3. Windows ("Microsoft Windows 10 Professionl 64-bit”または “Microsoft Windows 10 Enterprise 64-bit”)
   * + ダウンロード: <https://hub.docker.com/editions/community/docker-ce-desktop-windows>
     + 詳細は次のリンクをご参照ください。 <https://docs.docker.com/docker-for-windows/install/>
4. Linux をお使いになる場合は、下記のDockerをインストールして、下記のコマンドで実行できます。 (sudo systemctl start docker):
   * + Fedora Linuxの例です。 <https://docs.docker.com/install/linux/docker-ce/fedora/>
5. Dockerをスタートします。Docker Desktopを見るとDockerが実行されていることが確認できます。

macOS:  


Windows:  

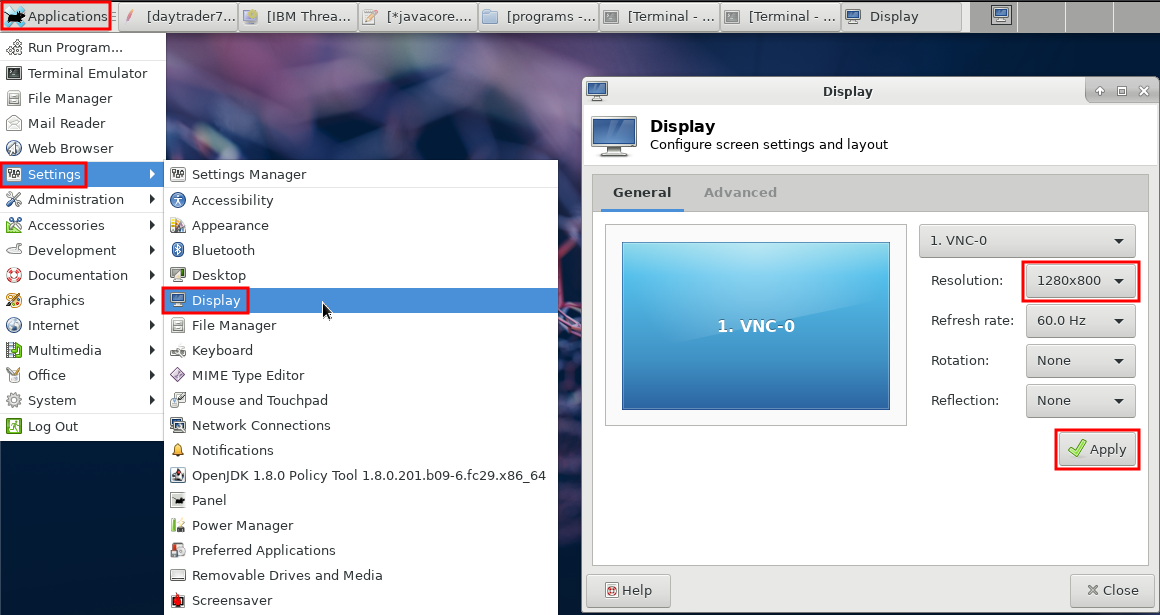

1. Dockerの実行環境に、十分なリソースを与えてください。特にメモリー使用量を十分に割り当ててください。
   1. Docker Desktopのアイコンをクリックし、 “Preferences…” (macOS) または “Settings” (Windows) を選びます。
   2. Advanced のタブを選択します。
   3. メモリー量を調節します。ワークショップを行うには、8GB以上のメモリーが必要です。
   4. “Apply & Restart”(macOS) または“Apply”(Windows) を押して設定を保存します。  
        
      macOS:  
        
        
      Windows:  
        
      
2. macOSやLinuxのターミナル、またはWindowsのコマンドプロンプトを開けます。  
   macOS:  
     
     
   Windows:  
   
3. Dockerのイメージを下記のコマンドでダウンロードします。

docker pull kgibm/fedorawasdebug

注）Dockerのイメージは20GBほどのサイズです。このDockerのイメージをワークショップの前に予めダウンロードされることをお勧めします。

1. ワークショップ用のDocker コンテナをスタートします。コマンドで使用されているポート(9080, 9443, 9043, 9081, 9444, 5901, 5902, 3389, 22, 9082, 9445)が、他のプログラムに使用されていないようにしてください。

docker run --cap-add SYS\_PTRACE --ulimit core=-1 --rm -p 9080:9080 -p 9443:9443 -p 9043:9043 -p 9081:9081 -p 9444:9444 -p 5901:5901 -p 5902:5902 -p 3390:3389 -p 22:22 -p 9082:9082 -p 9445:9445 -it kgibm/fedorawasdebug

1. コンテナの開始から30秒後くらいで VNCや Remote Desktopのサービスが立ち上がります。下記の要領で、Dockerコンテナにログインできることを確認してください。
   1. macOS 付属の VNCクライアント
      1. 新しいターミナルをあけて次のコマンドを実行します。
         1. open vnc://localhost:5902
         2. パスワードを入力してください: **websphere**
   2. Linux のVNC クライアント
      1. 新しいターミナルをあけて次のコマンドを実行します。
         1. vncviewer localhost:5902
         2. パスワードを入力してください: **websphere**
   3. Windows に VNCクライアントがインストールされている場合（Windows付属のソフトウェアではありません）
      1. 上記と同様に **localhost** のポート **5902** に接続し、パスワードとして **websphere**を入力してください。
   4. Windows リモートデスクトップ
      1. Windows のリモートデスクトップを使用する場合は、いくつかのステップが必要です。”2 補足”の[2.1 Windows のリモートデスクトップからDockerコンテナにアクセスする](#_Windows_のリモートデスクトップからDockerコンテナにアクセ) を参照してください。
   5. SSH:
      1. SSHやPuttyなどを使うと本番環境に近い状態でDocker コンテナにアクセスできます。ほとんどのワークショップのツールはGUIを使うので、VNCクライアント、またはリモートデスクトップで接続することが必要です。下記は、ターミナルからsshでログインするコマンド例です。
         1. ssh was@localhost
         2. Password: **websphere**
2. VNCクライアントから、ディスプレイの解像度をDocker コンテナの中で変更できます。VNCクライアントは自動的に新しい解像度に合わせて表示します。下記の例ではSettingからDisplay表示を1200 x 800に変えています。
3. 端末のブラウザ、またはリモートデスクトップやVNCのブラウザから<http://localhost:9080/> を指定してWAS Libertyにアクセスしてみましょう。
4. 同様に <http://localhost:9081/swat/> を指定してTraditional WAS で動いているアプリケーションにアクセスしてみます。
   1. Traditional WASはコンテナが起動してから数分で立ち上がります。
   2. Traditional WAS の管理コンソールには <https://localhost:9043/ibm/console> から下記のユーザー名とパスワードでログインできます。
      1. User: **wsadmin**
      2. Password: **websphere**
5. Dockerコンテナを起動して、VNCやリモートデスクトップで接続できることを確認したらワークショップの準備は完了です！

下記の手順にしたがって、コンテナを終了します。まず“docker ps”コマンドで、コンテナ名を見つけ、”docker stop”コマンドでコンテナを指定して終了します。コンテナが終了しても”docker images” コマンドで、ダウンロードしたワークショップのイメージが保存されていることが確認できます。ワークショップを始めるときは、このイメージを使って再びコンテナを起動します。

C:\>**docker ps**

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

928a1531600c kgibm/fedorawasdebug "/entrypoint.sh" About a minute ago Up About a minute 0.0.0.0:22->22/tcp, 0.0.0.0:3389->3389/tcp, 0.0.0.0:5901-5902->5901-5902/tcp, 0.0.0.0:9043->9043/tcp, 0.0.0.0:9080-9081->9080-9081/tcp, 0.0.0.0:9443-9444->9443-9444/tcp **keen\_brattain**

C:\>**docker stop keen\_brattain**

keen\_brattain

C:\>**docker ps**

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

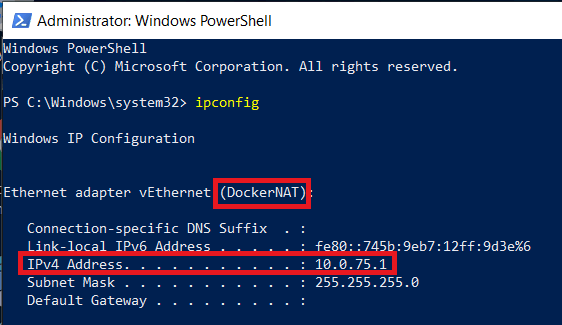
C:\>**docker images**

REPOSITORY TAG IMAGE ID CREATED SIZE

**kgibm/fedorawasdebug** latest 93ba47c73042 2 weeks ago 13.2GB

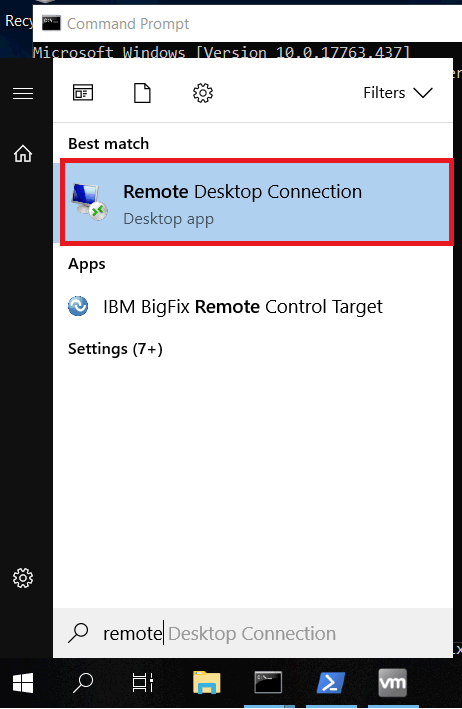
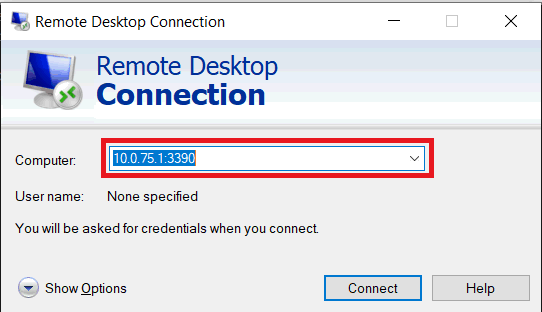
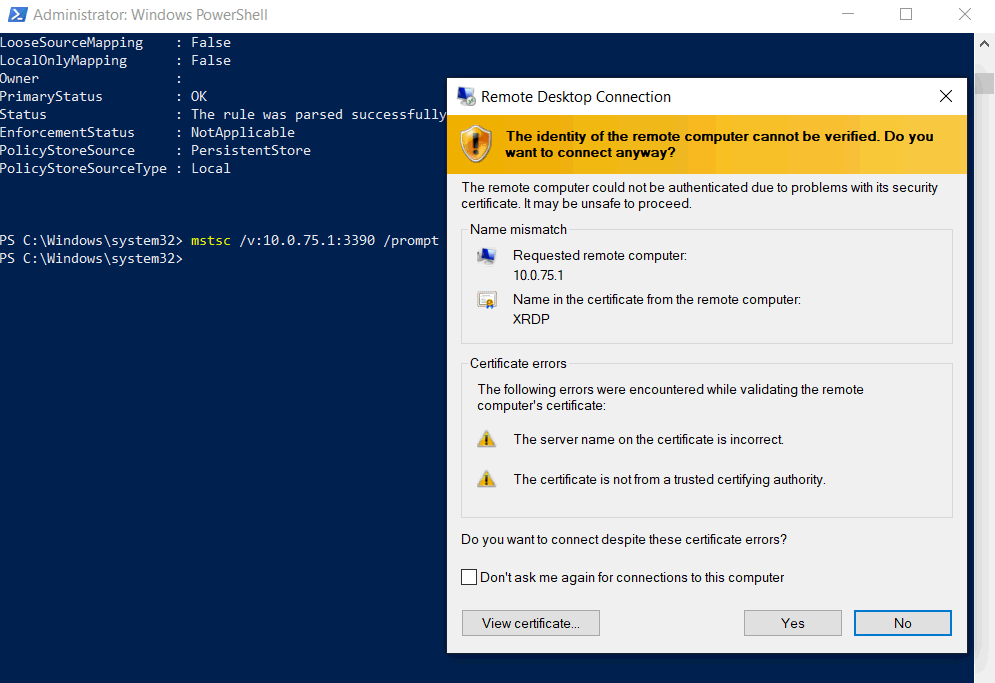
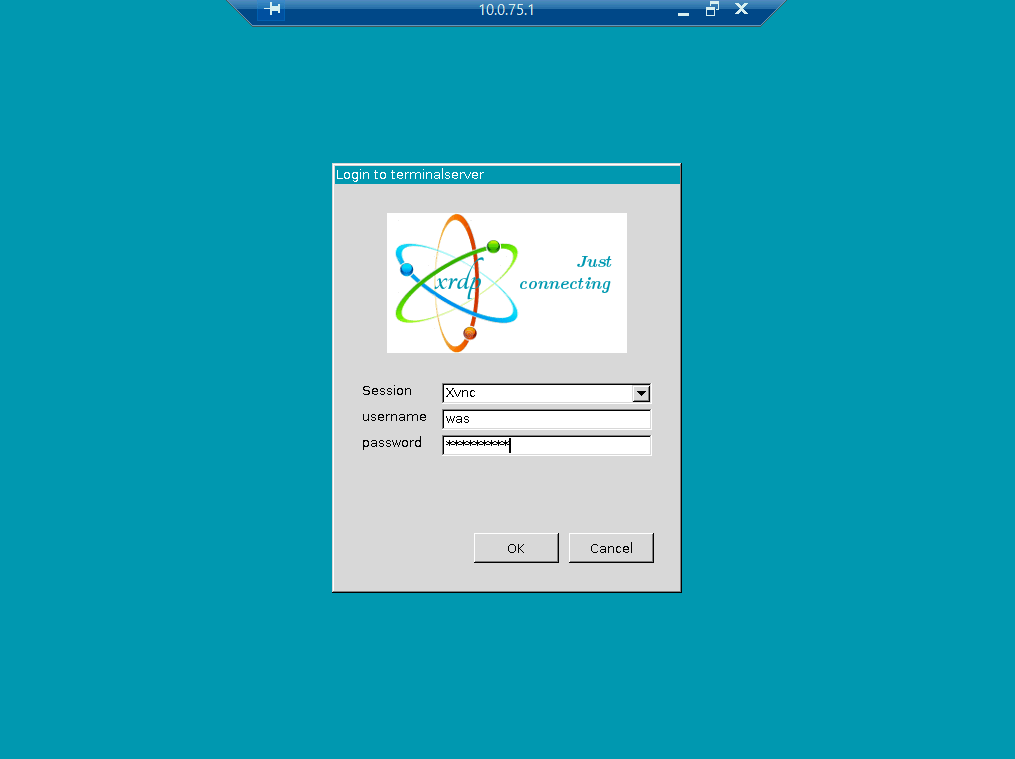
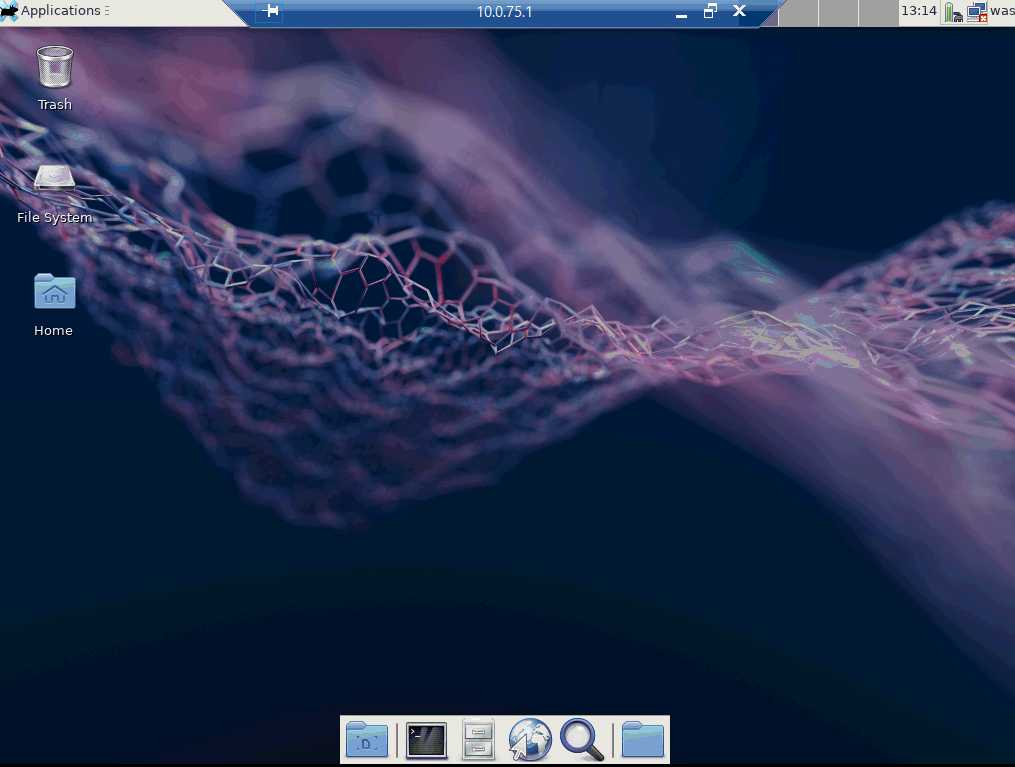
# 補足

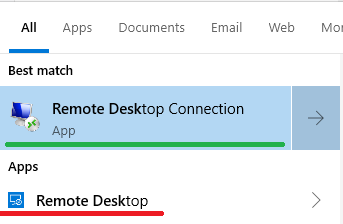
## Windows のリモートデスクトップからDockerコンテナにアクセスする

* 1. WindowsのPowerShell を管理者権限でオープンします。  
       
     
  2. ipconfigコマンドを実行し DockerNAT adapterのIPアドレスを見つけます。  
     
  3. 下記のコマンドをPowerShellから実行しDockerのポートがFirewallでも見えるようにRuleを作ります。

New-NetFirewallRule -Name "myRDP" -DisplayName "Remote Desktop Protocol" -Protocol TCP -LocalPort @(3389) -Action Allow

New-NetFirewallRule -Name "myContainerRDP" -DisplayName "RDP Port for connecting to Container" -Protocol TCP -LocalPort @(3390) -Action Allow

* 1. リモートデスクトップを起動します。  
     
  2. 先ほど項番2.で見つけた DockerNAT IP address (例: 10.0.75.1) にポート番号:3390 を指定して "Connect"を押します。  
       
     
  3. CertificateのWarningが出ますが、ワークショップ用のコンテナ接続で、同じ端末内なので "Yes" を押して接続します。  
       
     
  4. ユーザー名に **was** を、パスワードに **websphereを入力します。**  
       
     
  5. コンテナにリモートデスクトップ接続しました。  
       
     
  6. Windows10では、複数のリモートデスクトップのメニューが出ることがあります。動作するものを使ってください。[[1]](#footnote-1)



* 1. このワークショップでは、Dockerコンテナへのリモートデスクトップ接続を可能にするために、一般的なRDPポート(3389)をコンテナのポート(3390)にマップしています。[[2]](#footnote-2)

1. <https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/clients/remote-desktop-app-compare> [↑](#footnote-ref-1)
2. <https://social.msdn.microsoft.com/Forums/en-US/872129e4-07a5-48c3-86f7-996854e7a920/how-to-connect-via-rdp-to-container?forum=windowscontainers> [↑](#footnote-ref-2)