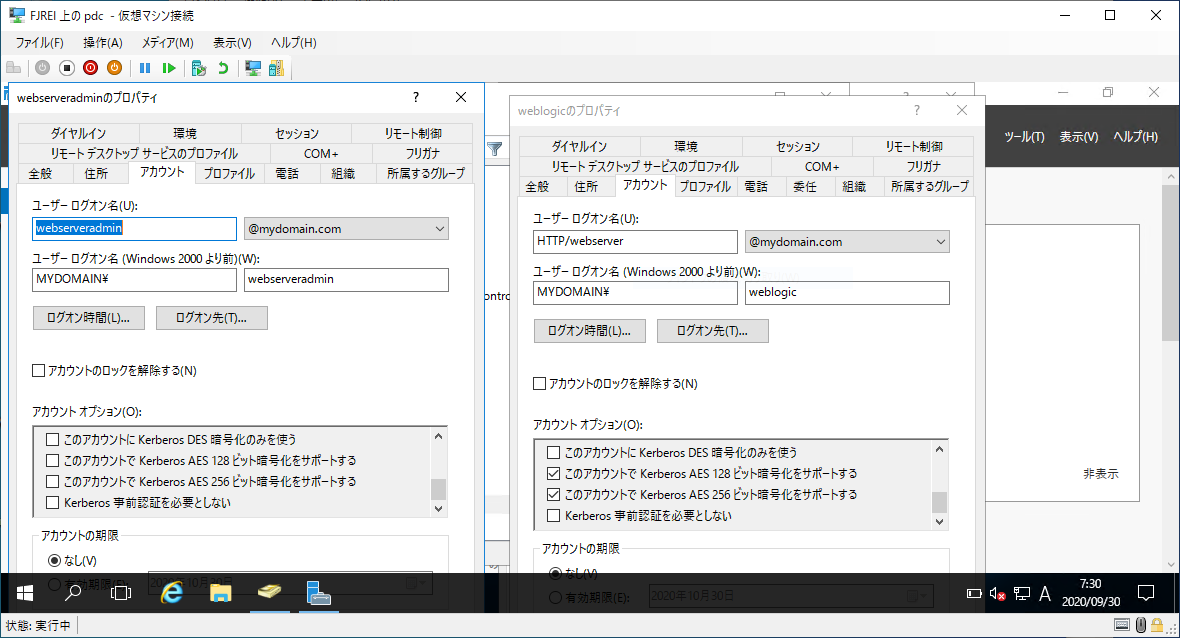
# SSOのセットアップ

* Web/APサーバーのADユーザー名
  + [:weblogic:]
* Web/APサーバーのWebLogicサーバーを表すサービス名
  + [:webserver:]
* ドメイン名
  + [:mydomain.com:]
* ドメインコントローラー
  + [:pdc.mydomain.com:]
* デプロイしたアプリケーション
  + [:webapp:]

## [pdc]ADユーザーの作成

ドメインコントローラーにて、Web/APサーバーを表すユーザー[:weblogic:]を作成します。



ADアカウントの状態

上の図において「weblogicのプロパティ」がWeb/APサーバーを表すユーザーです。 なお、「HTTP/webserver」となっている箇所は[:weblogic:]としてください(後述のktpassコマンドの実行によりHTTP/webserverに変更されます)。

### 注意点

* ADでのアカウントの種類が「コンピューター」ではなく「ユーザー」であることに注意してください。
* アカウント オプションでは、パスワードの有効期限を無期限に設定してください。
* アカウント オプションでは、「このアカウントで Kerberos AES 128ビット暗号化をサポートする」を有効にしてください。
* アカウント オプションでは、「このアカウントで Kerberos AES 256ビット暗号化をサポートする」を有効にしてください。
* アカウント オプションでは、DESの暗号化を無効にしてください。

## [webserver]krb.iniを作りWeb/APサーバーのC:\Windowsに配置する

Web/APサーバーに、以下の内容のC:\Windows\krb5.iniを作成してください。

[libdefaults]  
default\_realm = [:MYDOMAIN.COM:]  
ticket\_lifetime = 600  
allow\_weak\_crypto = true  
  
[realms]  
  
[:MYDOMAIN.COM:] = {  
kdc = [:pdc.mydomain.com:]  
admin\_server = [:pdc.mydomain.com:]  
default\_domain = [:MYDOMAIN.COM:]  
}  
  
[domain\_realm]  
.[:mydomain.com:] = [:MYDOMAIN.COM:]  
[:mydomain.com:] = [:MYDOMAIN.COM:]  
  
[appdefaults]  
autologin = true  
forward = true  
forwardable = true  
encrypt = true

## [webserver]C:\Windows\krb5.iniの内容の検証

C:\Windows\krb5.iniの内容が正しいことを確認するために、Web/APサーバーでOracle JDKに同梱されているkinitコマンドを以下のように実行してください。

> kinit [:weblogic:]  
Password for [:weblogic:]@[:MYDOMAIN.COM:]:  
New ticket is stored in cache file C:\Users\<実行中のユーザー>\krb5cc\_<実行中のユーザー名>

もしくは

> kinit [:weblogic:]@[:MYDOMAIN.COM:]  
Password for [:weblogic:]@[:MYDOMAIN.COM:]:  
New ticket is stored in cache file C:\Users\<実行中のユーザー>\krb5cc\_<実行中のユーザー名>

kinitコマンド実行の結果生成されるC:\Users\<実行中のユーザー>\krb5cc\_<実行中のユーザー名>は削除してください。

## [pdc]SPNの作成

ドメインコントローラーにて、ADユーザー[:weblogic:]に対してSPN(Service Provider Name)を割り当てます。

setspnコマンドのシンタックスは以下の通りです。

> setspn -S HTTP/<WebLogicサーバー名>@<ドメイン名大文字> <ADユーザー名>

setspnコマンドを以下のように実行してください。

> setspn -S HTTP/[:webserver:] [:weblogic:]  
> setspn -S HTTP/[:webserver:]@[:MYDOMAIN.COM:] [:weblogic:]  
> setspn -S HTTP/[:webserver:].[:mydomain.com:] [:weblogic:]  
> setspn -S HTTP/[:webserver:].[:mydomain.com:]@[:MYDOMAIN.COM:] [:weblogic:]

SPN割り当ての確認はsetspn -Lコマンドを使用します。

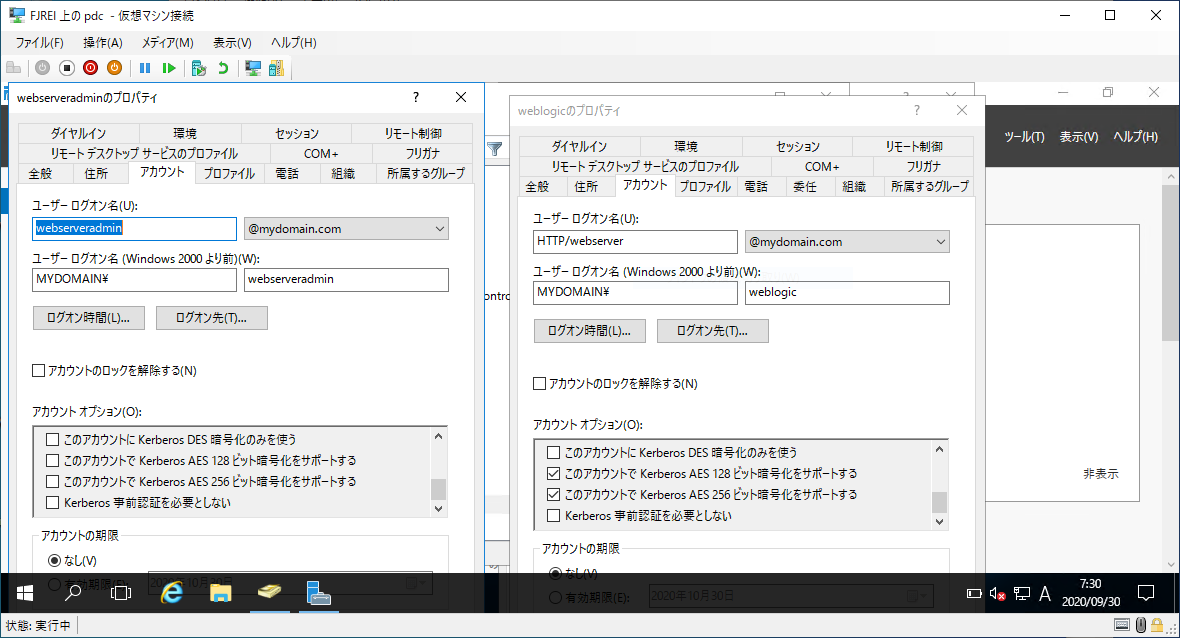
> setspn -L webserver  
次の項目に登録されている CN=[:weblogic:],CN=Users,DC=[:mydomain:],DC=[:com:]:  
 HTTP/[:webserver:].[:mydomain.com:]  
 HTTP/[:webserver:]  
 HTTP/[:webserver:]@[:MYDOMAIN.COM:]  
 HTTP/[:webserver:].[:mydomain.com:]@[:MYDOMAIN.COM:]

## [pdc]keytabファイルの作成とWeb/APサーバーへのコピー

ドメインコントローラーにて、keytabファイルを作成します。ktpassコマンドを以下のように実行してください。

> ktpass -princ HTTP/[:webserver:]@[:MYDOMAIN.COM:] -mapuser [:weblogic:] -pass [:password:] -crypto all -kvno 0 -ptype KRB5\_NT\_PRINCIPAL -out http-[:webserver:].[:mydomain.com:]-ktpass.keytab

ktpassコマンドの実行に成功した後、ADユーザー[:weblogic:]のログオン名がHTTP/[:webserver:]に変更されていることを確認してください。



ADアカウントの状態

## [webserver]keytabファイルの内容の検証

ドメインコントローラーで作成したhttp-[:webserver:].[:mydomain.com:]-ktpass.keytabを、 Web/APサーバーにコピーしてください。ここではD:\pp\weblogic\http-[:webserver:].[:mydomain.com:]-ktpass.keytabとします。

Web/APサーバーにて、http-[:webserver:].[:mydomain.com:]-ktpass.keytabファイルの内容の検証を行います。 klistコマンドを以下のように実行してください。

> klist -e -k http-[:webserver:].[:mydomain.com:]-ktpass.keytab

HTTP/[:webserver:]@[:mydomain.com:]のエントリーが(複数)存在していればkeytabファイルとして正常です。

## [webserver]krb5Login.confファイルの作成と配置

Web/APサーバーにて、以下の内容のkrb5Login.confを適当なフォルダに配置してください。 ここではD:\pp\weblogic\krb5Login.confとします。

com.sun.security.jgss.initiate {  
 com.sun.security.auth.module.Krb5LoginModule required principal="HTTP/[:webserver:]@[:MYDOMAIN.COM:]" useKeyTab="true" keyTab="D:/pp/weblogic/http-[:webserver:].[:mydomain.com:]-ktpass.keytab"  
 storeKey="true" debug="true";  
};  
com.sun.security.jgss.krb5.accept {  
 com.sun.security.auth.module.Krb5LoginModule required principal="HTTP/[:webserver:]@[:MYDOMAIN.COM:]" useKeyTab="true" keyTab="D:/pp/weblogic/http-[:webserver:].[:mydomain.com:]-ktpass.keytab"  
 storeKey="true" debug="true";  
};

krb5Login.confのパース処理はかなりパースエラーを起こしやすいため、改行の位置とスペースの数は上記に従って厳密に再現してください。

## [webserver]-Dパラメータの追加

D:\pp\Oracle\Middleware\Oracle\_Home\user\_projects\domains\base\_domain\startWebLogic.cmdに以下の1行を追加してください。

set JAVA\_OPTIONS=%JAVA\_OPTIONS% -Djavax.security.auth.useSubjectCredsOnly=false -Djava.security.auth.login.config=D:\pp\weblogic\krb5Login.conf -Djava.security.krb5.conf=C:\Windows\krb5.ini -Dweblogic.security.enableNegotiate=true -Dsun.security.krb5.debug=true

挿入位置は以下を参考にしてください。

@REM \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
@REM This script is used to start WebLogic Server for this domain.  
@REM   
@REM To create your own start script for your domain, you can initialize the  
@REM environment by calling @USERDOMAINHOME\setDomainEnv.  
@REM   
@REM setDomainEnv initializes or calls commEnv to initialize the following variables:  
@REM   
@REM BEA\_HOME - The BEA home directory of your WebLogic installation.  
@REM JAVA\_HOME - Location of the version of Java used to start WebLogic  
@REM Server.  
@REM JAVA\_VENDOR - Vendor of the JVM (i.e. BEA, HP, IBM, Sun, etc.)  
@REM PATH - JDK and WebLogic directories are added to system path.  
@REM WEBLOGIC\_CLASSPATH  
@REM - Classpath needed to start WebLogic Server.  
@REM PATCH\_CLASSPATH - Classpath used for patches  
@REM PATCH\_LIBPATH - Library path used for patches  
@REM PATCH\_PATH - Path used for patches  
@REM WEBLOGIC\_EXTENSION\_DIRS - Extension dirs for WebLogic classpath patch  
@REM JAVA\_VM - The java arg specifying the VM to run. (i.e.  
@REM - server, -hotspot, etc.)  
@REM USER\_MEM\_ARGS - The variable to override the standard memory arguments  
@REM passed to java.  
@REM PRODUCTION\_MODE - The variable that determines whether Weblogic Server is started in production mode.  
@REM DERBY\_HOME - Derby home directory.  
@REM DERBY\_CLASSPATH  
@REM - Classpath needed to start Derby.  
@REM   
@REM Other variables used in this script include:  
@REM SERVER\_NAME - Name of the weblogic server.  
@REM JAVA\_OPTIONS - Java command-line options for running the server. (These  
@REM will be tagged on to the end of the JAVA\_VM and  
@REM MEM\_ARGS)  
@REM PROXY\_SETTINGS - These are tagged on to the end of the JAVA\_OPTIONS. This variable is deprecated and should not  
@REM be used. Instead use JAVA\_OPTIONS  
@REM   
@REM For additional information, refer to "Administering Server Startup and Shutdown for Oracle WebLogic Server"  
@REM \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
set JAVA\_OPTIONS=%JAVA\_OPTIONS% -Djavax.security.auth.useSubjectCredsOnly=false -Djava.security.auth.login.config=D:\pp\weblogic\krb5Login.conf -Djava.security.krb5.conf=C:\Windows\krb5.ini -Dweblogic.security.enableNegotiate=true -Dsun.security.krb5.debug=true  
set SCRIPTPATH=%~dp0  
set SCRIPTPATH=%SCRIPTPATH%  
for %%i in ("%SCRIPTPATH%") do set SCRIPTPATH=%%~fsi

## [webserver]WebLogicコンソールでのセキュリティ・レルムの設定

グローバルなセキュリティ・レルムmyrealmを追加します。

WebLogicコンソールにログインし、左ペインからbase\_domain→セキュリティ・レルムを開いてください。

新規ボタンを押下してください。

新しいレルムの名前をmyrealmとしてください。

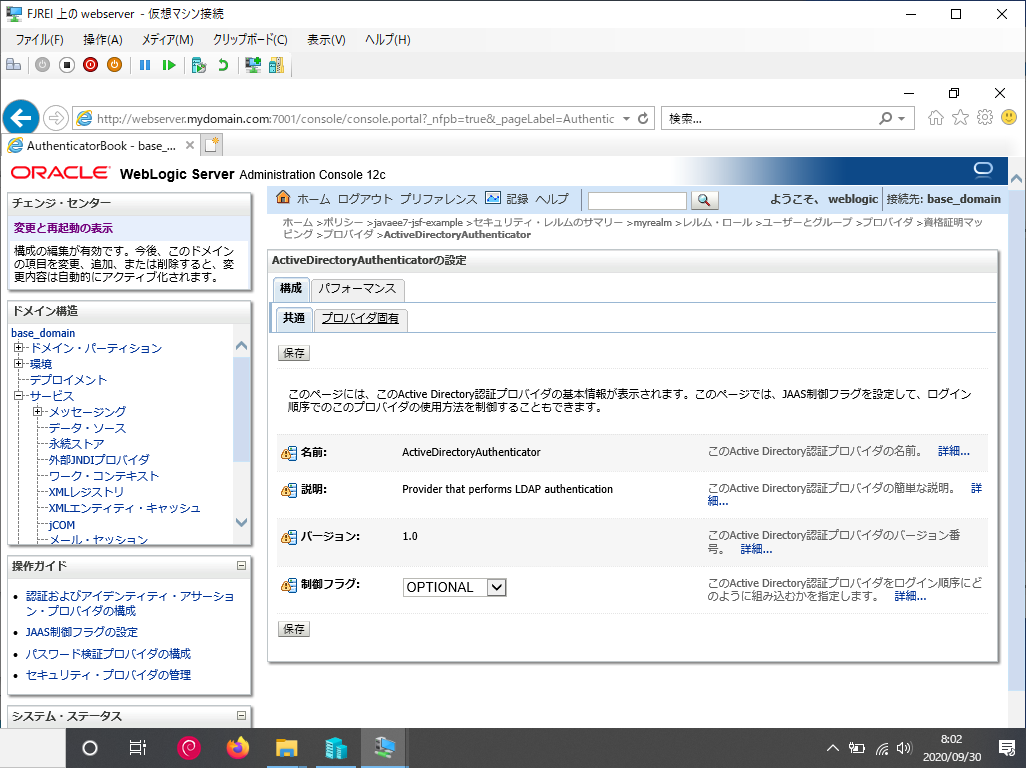
myrealmの設定を開いて、以下の通りの設定内容であることを確認してください。

myrealmのプロバイダタブを開き、Active Directoryプロバイダを追加してください。 名称はActiveDirectoryAuthenticator、タイプはActiveDirectoryAuthenticatorとしてください。

次に、myrealmのプロバイダタブからNegotiateIdentityAsserterプロバイダを追加してください。 名称はSPNEGO、タイプはNegotiateIdentityAsserterとしてください。

## [webserver]Active Directory Providerの設定

制御フラグはOPTIONALに設定します。



ActiveDirectoryAuthenticatorの共通タブの設定

ActiveDirectoryAuthenticatorのプロバイダ固有タブを開き、以下の通り設定してください。

* ホスト
  + [:pdc.mydomain.com:]
* ポート
  + 389
* プリンシパル
  + [:weblogic:]
* 資格証明
  + <[:weblogic:]のパスワード>
* 資格証明の確認
  + <[:weblogic:]のパスワード>

「ユーザー」セクションでは以下の通り設定してください。

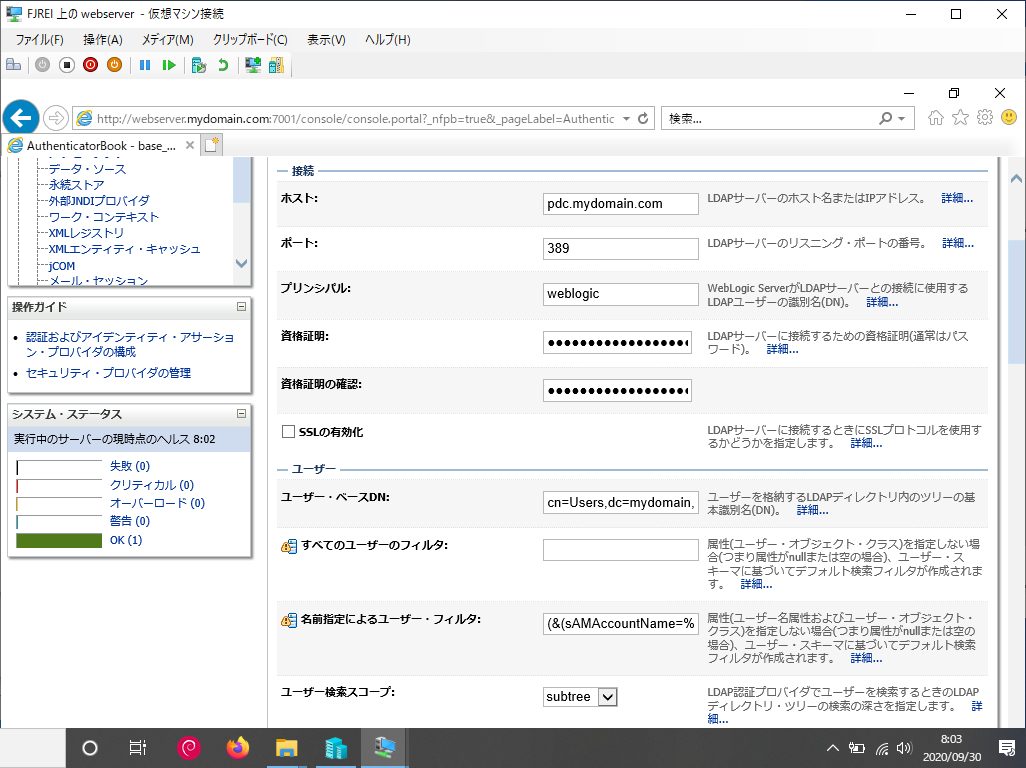
* ユーザー・ベースDN
  + cn=Users,dc=[:mydomain:],dc=[:com:]
  + ユーザーを管理するロケーションに合わせて変更してください。
* 名前指定によるユーザー・フィルタ
  + (&(sAMAccountName=%u)(objectclass=user))
* ユーザー名属性
  + sAMAccountName
* ユーザーオブジェクトクラス
  + user

「グループ」セクションでは以下の通り設定してください。

* グループ・ベースDN
  + cn=Users,dc=[:mydomain:],dc=[:com:]
* 名前指定によるグループ・フィルタ
  + (&(cn=%g)(objectclass=group))

「静的グループ」セクションでは以下の通り設定してください。

* 静的グループ名属性 cn
* 静的グループ・オブジェクト・クラス
  + group
* 静的メンバーDN属性 member
* メンバーDN指定による静的グループDNフィルタ
  + (&(member=%M)(objectclass=group))

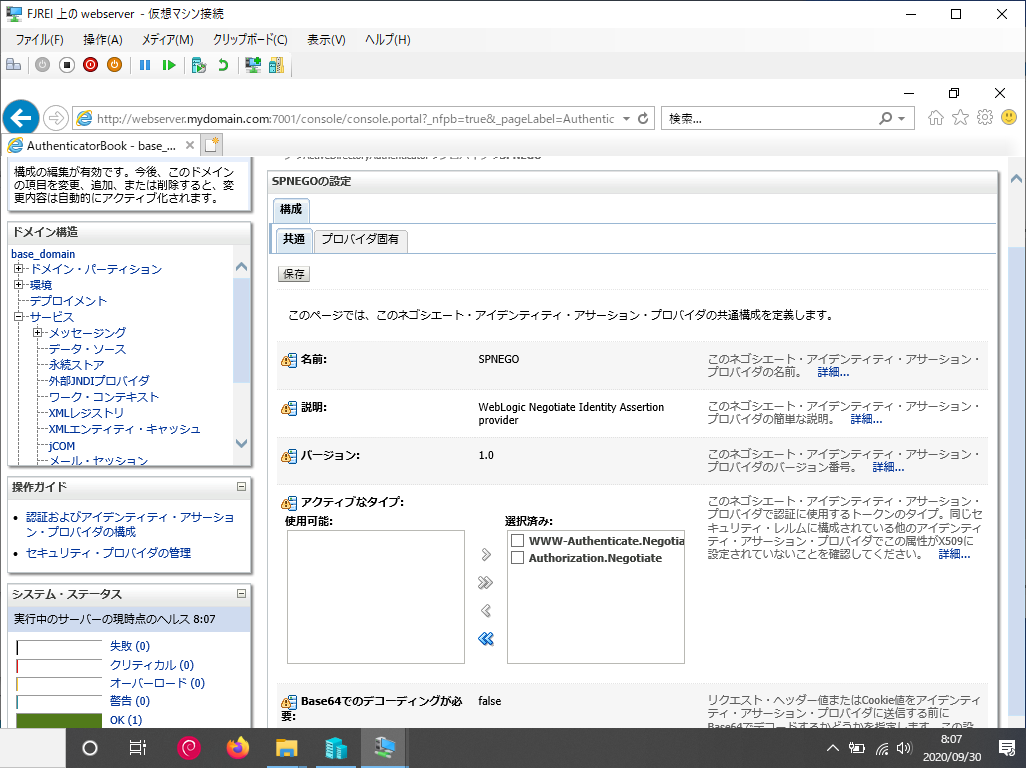


ActiveDirectoryAutuenticatorのプロバイダ固有の設定

設定後、保存ボタンを押下してください。

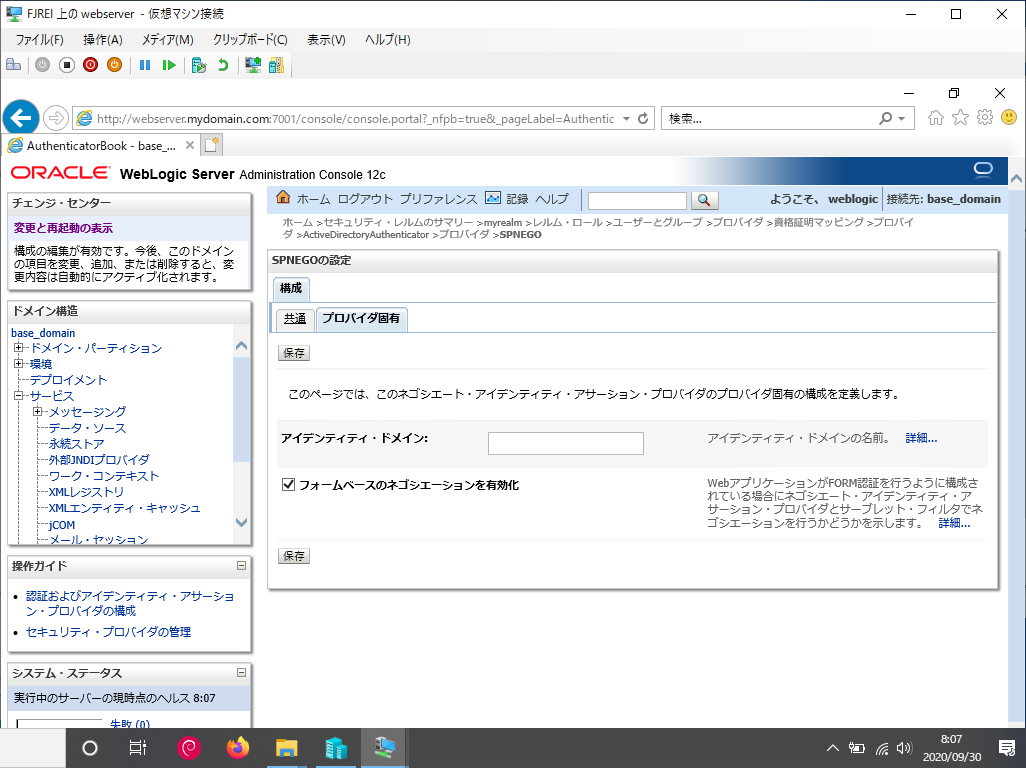
## [webserver]NegotiateIdentityAsserterの設定

共通タブを開き、アクティブなタブの「WWW-Authenticate.Negotiate」と「Authorization.Negotiate」を使用可能欄から選択済みへと移動してください。



NegotiateIdentityAsserterの設定

プロバイダ固有タブを開き、以下の通りに設定してください。



SPNEGOの設定

設定後、保存ボタンを押下してください。

## [webserver]Webアプリケーションのweblogic.xmlの設定

デプロイメントされた[:webapp:]のweblogic.xmlに対して、以下のように<security-role-assignment>要素を追加します。

<?xml version="1.0" encoding="UTF-8"?>  
<weblogic-web-app xmlns="http://xmlns.oracle.com/weblogic/weblogic-web-app"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://xmlns.oracle.com/weblogic/weblogic-web-app http://xmlns.oracle.com/weblogic/weblogic-web-app/1.0/weblogic-web-app.xsd">  
 <security-role-assignment>  
 <role-name>operator</role-name>  
 <principal-name>operators</principal-name>  
 </security-role-assignment>  
</weblogic-web-app>

…

## [webserver]web.xmlの設定

デプロイメントされた[:webapp:]のweb.xmlに対して、以下のように<security-constraint>要素、<login-config>要素、<security-role>要素を追加します。

<security-constraint>  
 <display-name>Operator Access</display-name>  
 <web-resource-collection>  
 <web-resource-name>SampleTestServlet</web-resource-name>  
 <url-pattern>/faces</url-pattern>  
 </web-resource-collection>  
 <auth-constraint>  
 <role-name>operator</role-name>  
 </auth-constraint>  
 </security-constraint>  
  
 <login-config>  
 <auth-method>CLIENT-CERT</auth-method>  
 </login-config>  
  
 <security-role>  
 <description>security-role operator</description>  
 <role-name>operator</role-name>  
 </security-role>