ORACLE

Microsoft Active Directory ネーミングのためのOracle Databaseクライアントの構成

Microsoft Active Directoryにおけるネットワークの名前およびアドレスの一元化

2023年6月

目次

1. はじめに

アプリケーションをOracle Databaseに接続する際は、データベースのホスト名やサービス名などの情報を含む接続記述子を使用することが必要です。接続記述子は、アプリケーションによってさまざまな方法で指定できます。たとえば、アプリケーション接続リクエストでハードコードしたり、tnsnames.ora構成ファイルに保存された接続記述子にマッピングされる識別子をアプリケーションが渡したりすることができます。tnsnames.oraファイルを使用する代わりに、外部のマッピング・サービスを使用して接続記述子を検索することもできます。提供されるサービスの1つがディレクトリ・ネーミングです。

ディレクトリ・ネーミングによって、ネットワークの名前とアドレスが1つの場所で一元化され、名前の変更と更新が簡単に管理できるようになります。これにより、tnsnames.oraファイルに保存された接続記述子を管理者が変更する必要がなくなります。大規模な組織には、何百、何千ものデータベース・アプリケーションやtnsnames.oraファイルが存在する可能性があります。

このドキュメントでは、Microsoft Active Directory（AD）から名前解決を実現するために必要な構成手順について説明します。

2. Oracle NetネーミングのためのActive Directoryスキーマの拡張

ADスキーマは、Oracle Databaseクライアント・ネーミング固有のスキーマ属性を使用して拡張する必要があります。このスキーマ拡張は、AD DomainまたはAD Domain Forestあたり1回のみ実行する必要があります。

AD Domainの一部であるWindowsクライアント・マシンは、ネイティブ・バインディングによってディレクトリ・ネーミングを使用するように構成できます（すなわち、ADにログインするためにオペレーティング・システム資格証明を使用）。

ADスキーマを拡張するには以下を実行します。

1. Windows Domainクライアント・マシンのいずれか、またはActive Directoryサーバー上で、Active Directory管理者としてOracle Net Configuration Assistantツール（“Oracle NetCA”）を実行します。Oracle Database Clientは最初にインストールする必要があります。

「Start」メニューを開き、「Oracle <home name>」から「Net Configuration Assistant」を選択してOracle NetCAを実行します。

2. 「Active Directory in Directory Usage」構成画面を開きます。
3. スキーマ作成オプションを選択します。
4. Active Directoryサーバーの詳細を入力します。

正常に完了すると、ADを使用してOracle Netネーミング・オブジェクトおよび属性を保存できるようになります。

3. Active Directoryネーミング用クライアントのセットアップ

ディレクトリ・ネーミングを使用するためのクライアント構成ファイルを作成するには、Oracle NetCAを使用します。このツールを使用すると、OracleContextオブジェクトを含むネーミング・コンテキストを選択できます。すべてのディレクトリ・ネーミング・オブジェクトは、OracleContextの下に作成されます。

構成を開始するには以下を実行します。

1. 前述のようにStartメニューからOracle NetCAを起動
2. 「Directory Usage Configuration」オプションを選択
3. Directory Typeドロップダウンで「Active Directory」を選択
4. ADサーバーの詳細を入力
5. ネーミング・コンテキストを選択

完了すると、Oracle NetCAによってOracle home network\adminサブディレクトリにファイルldap.oraが作成されます。

以下はldap.oraファイルの例です。

```
DEFAULT_ADMIN_CONTEXT = "DC=example,DC=com"
DIRECTORY_SERVER_TYPE = AD
```

ディレクトリ・ネーミングを使用するクライアント・マシンすべてにldap.oraファイルをコピーするか、またはそれらすべてのマシンでOracle NetCAを再度実行します。

ご参考までに、Oracle Net Configuration Assistantドキュメントはこちらにあります。

4. ディレクトリ・サーバーでのエントリの作成

ディレクトリ・サーバーでネーミング・エントリを作成して管理するには、最初に1つのディレクトリ・ユーザーを作成する必要があります。このユーザーが、上記でセットアップしたOracleContextに保存されるネーミング・エントリを管理します。ユーザーを作成するには、ディレクトリ・サーバー・ドキュメントに従ってください。以下の例では、名前を"mynaminguser"と仮定します。

そして、Oracle Net Managerツール（“Oracle NetManager”）、またはコマンドライン・ツールを使用することにより、以降のセクションに示すネーミング・エントリを作成して管理できます。

4.1 Oracle Net Managerによるネーミング・エントリの作成

次の手順を実行してください。

1. 「Start」メニューを開き、Configuration and Migration Toolsから「Net Manager」を選択してOracle NetManagerを実行します。

   セクション3で作成されたldap.oraファイルがインプレースである場合は、Oracle NetManagerによってディレクトリ・アイコンがツリー・ナビゲータに表示されます。

ディレクトリ・サブツリーがない場合は、セクション4.2のコマンドライン・ツールを使用してください。

2. ディレクトリ・サブツリーをクリックして展開します。

   ディレクトリ・サーバー・ユーザーの資格証明を入力するように求められます。セクション4の開始時に作成された資格証明を入力します。

3. サービス・ネーミング・サブツリーを選択します。

4. 左側のメニューにある緑色のプラス記号アイコンをクリックします。

5. ネット・サービス名ウィザード・プロンプトに従って、ネット・サービス名および記述子をディレクトリ・サーバーに作成します。

詳しくは、Oracle Net Managerドキュメントを参照してください。

4.2 コマンドライン・ツールを使用したネーミング・エントリの作成

Oracle Net Managerに代わる方法として、ldapaddやldapdeleteなどのLDAPコマンドライン・ツールを使用する方法があります。

たとえば、エイリアス“sales”を追加するには、以下に示すようにファイルsales.ldifを作成します。

```
dn: cn=sales,cn=oraclecontext,dc=example,dc=com
objectclass: top
objectclass: orclNetService
orclnetdescstring:
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=databasehost)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=mydbservice.us.example.com)))
cn: sales
```

この例では、OracleContextが“dc=example,dc=com”の下にあることを前提としています。Oracle Databaseのホスト、ポート、サービス名を使用するようにorclNetDescStringを調整します。

そして、以下の構文を使用してldapaddを実行します。

```
ldapadd    -h <ldap host> -p <ssl port> -D <Bind DN> -U 2 -q \
           -Q -W file:<wallet directory> -P <wallet password> \
           -f <ldif file>
```

たとえば、次のとおりです。

```
ldapadd    -h mydirectoryserverhost -p 636 \
           -D "cn=mynaminguser,dc=example,dc=com" -U 2 -q \
           -Q -W file:C:\oracle\wallets -P MySecret -f sales.ldif
```

双方向のTLS 1.2認証を使用している場合は、代わりにオプション-U 3を使用する必要があります。

エントリを削除するには、ldapdeleteコマンドを使用します。構文はldapaddに似ていますが、削除するエイリアスを以下のように指定します。

```
ldapdelete     -h <ldap host> -p <ssl port> -D <Bind DN> -U 2 -q \
               -Q -W file:<wallet directory> -P <wallet password> \
               <alias DN to be deleted>
```

または、削除する1つまたは複数のDNを含むファイルを以下のように指定できます。

```
ldapdelete     -h <ldap host> -p <ssl port> -D <Bind DN> -U 2 -q \
               -Q -W file:<wallet directory> -P <wallet password> \
           -f <ldif file with DNs to be deleted>
```

たとえば、次のとおりです。

```
ldapdelete    -h mydirectoryserverhost -p 636 \
              -D "cn=mynaminguser,dc=example,dc=com" -U 2 -q \
              -Q -W file:C:\oracle\wallets -P MySecret \
              "cn=sales,cn=oraclecontext,dc=example,dc=com"
```

または

```
ldapdelete    -h mydirectoryserverhost -p 636 \
              -D "cn=mynaminguser,dc=example,dc=com" -U 2 -q \
              -Q -W file:C:\oracle\wallets -P MySecret \
              -f sales-delete.ldif
```

この例では、sales-delete.ldifに以下のような行が含まれます。

```
cn=sales,cn=oraclecontext,dc=example,dc=com
```

ldapaddのように、双方向のTLS 1.2認証にはオプション-U 3を使用できます。

5. ディレクトリ・ネーミングのためのクライアント側の構成

クライアント・マシンの構成を継続します。

5.1 ネーミング・メソッド参照の有効化

ディレクトリ・ネーミングを使用するすべてのマシン上で、sqlnet.ora構成ファイルを作成または編集します。このファイルは、セクション3のldap.oraファイルと同じディレクトリにある必要があります。

sqlnet.oraファイルで、LDAPをNAMES.DIRECTORY_PATHパラメータに追加します。アプリケーション接続記述子は、そのパラメータで指定したネーミング・メソッドの順に評価されます。たとえば、最初にLDAPを試行してからEasy Connect構文を使用してフォールバックし、最後にtnsnames.oraファイルで接続文字列を参照する場合の例は次のとおりです。

```
NAMES.DIRECTORY_PATH = (LDAP, EZCONNECT, TNSNAMES)
```

NAMES.DIRECTORY_PATHエントリが存在しない場合、LDAPは使用されますが最初には考慮されません。

5.2 ディレクトリ・サーバーの認証

デフォルトでは、ディレクトリ・ネーミングで匿名バインディングが実行されます。ただし、ディレクトリ・サーバーが匿名バインディングを無効化している場合は、以下に示す2つのうちいずれかの方法で、ディレクトリ・サーバーに対して認証を構成する必要があります。

5.2.2 Active Directory Windowsネイティブ認証

Active Directoryでは、データベース・クライアントはディレクトリ認証にWindowsログイン資格証明を使用できます。

たとえば、以下の行をsqlnet.oraに追加します。

```
NAMES.DIRECTORY_PATH = (LDAP, TNSNAMES, EZCONNECT)
NAMES.LDAP_AUTHENTICATE_BIND = TRUE
```

5.2.3 ユーザー名およびパスワードベースの認証

または、ネイティブ認証を使用する代わりに、ユーザー名およびパスワードを使用してADサーバーにアクセスできます。この認証方式は、ADを使用するMicrosoft Windows以外のアプリケーションでも使用できます。アプリケーションは、Oracle Database 21c（またはそれ以降）のクライアント・ライブラリを使用する必要があります。

この方式では、ディレクトリ・サーバーのユーザー名およびパスワードはウォレットに保存されます。クライアントは、これらの資格証明を一方向のTLS接続経由でディレクトリ・サーバーへ渡すことによって認証を行います。

sqlnet.oraファイルを編集し、認証されたバインディングを有効化してバインド方式を設定します。たとえば、以下の行をファイルに追加します。

```
NAMES.LDAP_AUTHENTICATE_BIND = TRUE
NAMES.LDAP_AUTHENTICATE_BIND_METHOD = LDAPS_SIMPLE_AUTH
WALLET_LOCATION =
        (SOURCE = (METHOD = FILE)
            (METHOD_DATA =
            (DIRECTORY = <wallet directory>)
        )
```

ldap.oraファイルを編集してディレクトリ・サーバーのアドレスとポートを含む行を追加します。

```
DIRECTORY_SERVERS = (<domain controller name>:<clear text port>:
<TLS port>)
```

たとえば、次のとおりです。

```
DIRECTORY_SERVERS = (myadserver.mycompany.com:389:636)
```

ディレクトリ・サーバー証明書のルートCA証明書を取得し、以下のようにウォレットに保存します。ユーザー名とパスワードも追加する必要があります。

1. ウォレットがまだ存在しない場合は作成します。

   ```
   orapki wallet create \
         -wallet <directory to create the wallet in>
   ```

2. 証明書をウォレットに追加します。

   ```
   orapki wallet add -wallet <wallet directory> \
         -trusted_cert -cert <root CA certificate>
   ```

3. ユーザー名を追加します。

   ```
   mkstore -wrl <wallet directory> -createEntry \
         oracle.ldap.client.dn <DN of the user>
   ```

   たとえば、次のとおりです。

   ```
   mkstore -wrl /app/wallet -createEntry \
         oracle.ldap.client.dn "cn=user1,dc=acme,dc=com"
   ```

   Active Directoryでは、DNユーザー名は、User Principal NameまたはDown Level Logon Name（別名SAMAccountName）となることもあります。

   たとえば、fooがドメイン名でuser1がユーザー名の場合は次のとおりです。

   ```
   mkstore -wrl C:\oracle\wallets -createEntry \
         oracle.ldap.client.dn "foo\user1"
   ```

   または

   ```
   mkstore -wrl C:\oracle\wallets -createEntry \
   ```

```
        oracle.ldap.client.dn "user1@foo"
```

4. ユーザーのパスワードを追加します。

```
mkstore -wrl <wallet directory> -createEntry \
        oracle.ldap.client.password <password>
```

5. ウォレットを自動ログオン・ウォレットにします。

```
orapki wallet create -wallet <wallet directory> \
        -auto_login
```

6. 接続のテスト

ネーミング・サーバーの構成が完了しました。構成したクライアント・マシンのいずれかからSQL*Plusを実行することによって接続を検証できます。ネットワーク構成ファイルがデフォルトの場所にあること、または環境変数TNS_ADMINをそれらのファイルが含まれるディレクトリに設定していることを確認してください。データベース・ユーザー“scott”が存在し、エイリアス“sales”がsales.ldifファイルにある場合は、次のようにして接続します。

```
sqlplus scott@sales
```

7. まとめ

この技術概要では、Microsoft Active Directoryネーミング用にOracle Databaseクライアントを構成する方法について説明しました。