



## **AOUG Hands-On Tag**

Oracle Centrally Managed User 18c/19c

**15 Mai 2019, Version 1**

*Trivadis AG  
Sägereistrasse 29  
8152 Glattbrugg  
info@trivadis.com  
+41 58 459 55 55*

## Inhalt

<b>Einleitung Oracle Centrally Managed User 18c/19c</b>	<b>4</b>
<b>Centrally Managed User 18c</b>	<b>5</b>
Active Directory Konfiguration . . . . .	5
Server und Datenbank Konfiguration . . . . .	8
Benutzer und Rollen . . . . .	11
Zusatzaufgabe: Rollen und Administratoren . . . . .	12
Service Principle und Keytab Datei . . . . .	13
SQLNet Konfiguration . . . . .	15
Kerberos Authentifizierung . . . . .	16
Zusatzaufgaben . . . . .	17
<b>Übungen: Datenbank Authentifizierung und Password Verifier</b>	<b>18</b>
Überprüfung der aktuellen Password Verifier . . . . .	18
Anpassen der Password Verifier . . . . .	20
Zusatzaufgaben . . . . .	22
<b>Übungen: Kerberos Authentifizierung</b>	<b>22</b>
Service Principle und Keytab Datei . . . . .	22
SQLNet Konfiguration . . . . .	24
Kerberos Authentifizierung . . . . .	25
Zusatzaufgaben . . . . .	26
<b>Übungen: Centrally Managed User 18c</b>	<b>26</b>
Active Directory . . . . .	26
Server und Datenbank Konfiguration . . . . .	29
Benutzer und Rollen . . . . .	30
Zusatzaufgabe: Rollen und Administratoren . . . . .	32
<b>Übungen: Oracle Unified Directory</b>	<b>34</b>
Oracle Unified Directory Server Instanz . . . . .	34
Oracle Unified Directory Proxy Instanz . . . . .	35
Oracle Unified Directory Services Manager . . . . .	37
Zusatzaufgaben: Administration, Hochverfügbarkeit und Backup & Recovery . . . . .	38
<b>Übungen: Oracle Enterprise User Security</b>	<b>39</b>
SQLNet Konfiguration Enterprise User Security . . . . .	40
Enterprise User Security Datenbank Konfiguration . . . . .	40

Oracle Enterprise User Security Mappings . . . . .	40
Oracle Enterprise User Security Rollen . . . . .	42
Oracle Enterprise User Security Proxy Authentifizierung . . . . .	45
<b>Demo- und Lab Umgebung</b>	<b>46</b>
Architektur . . . . .	46
Oracle Datenbank Server . . . . .	47
Oracle Unified Directory Server . . . . .	49
MS Active Directory Server . . . . .	51
<b>Links und Referenzen</b>	<b>54</b>
OUD EUS Workshop . . . . .	54
Oracle Dokumentation . . . . .	54
Software und Tools . . . . .	54

## Einleitung Oracle Centrally Managed User 18c/19c

Mit der Live-Demo Oracle Centrally Managed User 18c/19c wird die Active Directory Integration am Beispiel der Trivadis Demo und Lab Umgebung aufgezeigt. Die Live-Demo wird am AOUG Hands-On Tag im Rahmen der Anwenderkonferenz 2019 durchgeführt. Die Testumgebung besteht, wie man in der folgenden Abbildung sehen kann, jeweils aus drei virtuellen Systemen.

- DB Server mit Oracle 19c
- Windows Server 2016 mit MS Active Directory
- OUD Server mit OUD 12.2.1.3.0 für die LDAP Namensauflösung

Optional können für weitere Tests zusätzliche VM's mit Oracle 18c, 12c oder 11g genutzt werden.

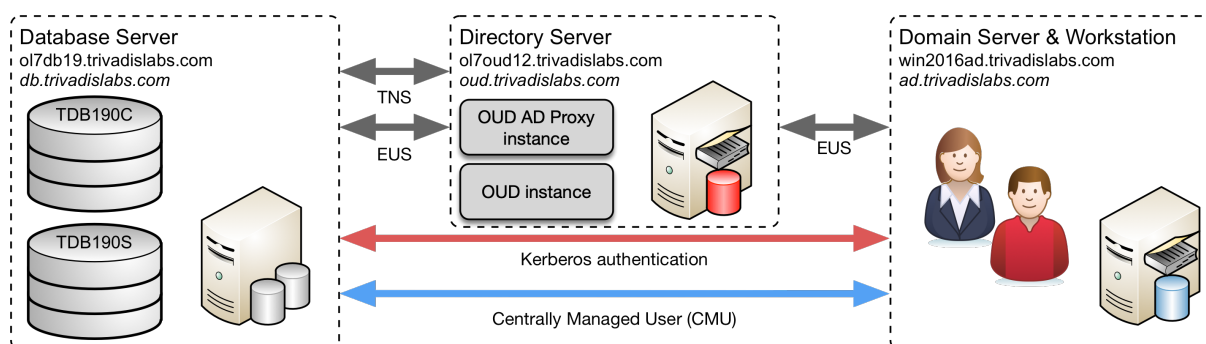


Abb. 1: Architektur Lab Umgebung

Wichtigsten Login Informationen im Überblick:

- Datenbank Server (Linux VM)
  - **Host Name** : ol7db19.trivadislabs.com
  - **Host Alias**: db.trivadislabs.com
  - **Interne IP Adresse** : 10.0.0.19
- Directory Server (Linux VM)
  - **Host Name** : ol7oud12.trivadislabs.com
  - **Host Alias**: oud.trivadislabs.com
  - **Interne IP Adresse** : 10.0.0.5
- Active Directory Server (Windows VM)
  - **Host Name** : win2016ad.trivadislabs.com
  - **Host Alias**: ad.trivadislabs.com
  - **Interne IP Adresse** : 10.0.0.4
- Benutzer und Passwörter
  - root / gemäss Referent oder SSH Key
  - oracle / gemäss Referent oder SSH Key
  - sys / manager

- system / manager
- TRIVADISLABS\Administrator / gemäss Referent
- Allgemein AD User ist Nachname/LAB01schulung

Das Login erfolgt jeweils via vagrant respektive ssh. Alternativ kann man auch direkt mit SSH oder Putty auf den weitergeleiteten Port zugreifen.

```
vagrant ssh  
sudo su - oracle
```

Im Kapitel [Demo- und Lab Umgebung](#) wird die Testumgebung etwas ausführlicher beschrieben. Zusätzlich besteht die Möglichkeit, selber eine eingene Testumgebung aufzubauen. Hierzu gibt es ein GitHub Repository [oehrli/trivadislabs.com](https://github.com/oehrli/trivadislabs.com) mit entsprechender Dokumentation, Scripts, Vagrant Files etc. um die Trivadis LAB Umgebung basierend auf Oracle [Virtualbox](#) und [vagrant](#) nahezu vollautomatisch lokal aufzubauen.

## Centrally Managed User 18c

**Ziele:** Konfiguration von Centrally Managed Users für die Datenbank TDB180S. Erweitern des Active Directory Schemas inklusive der Installation des Password Filter Plugins. Erstellen von Mappings für Benutzer und Rollen sowie erfolgreichem Login mit Passwort Authentifizierung.

### Active Directory Konfiguration

Arbeitsumgebung:

- **Server:** win2016ad.trivadislabs.com
- **Benutzer:** Administrator

Die folgenden Arbeiten werden in der Regel in Zusammenarbeit mit dem Windows respektive Active Directory Administrator durchgeführt. Je nach Unternehmensgrösse sind allenfalls noch weitere IT Bereiche mit involviert.

Für das Oracle Wallet wird das Root Zertifikat vom Active Directory Server benötigt. Dieses kann in der LAB Umgebung einfach via Commandline exportiert werden. Dazu öffnet man ein Command Prompt (`cmd.exe`) und exportiert das Root Zertifikat. Das exportierte Root Zertifikat muss anschliessend mit WinSCP auf den Datenbank Server in das Verzeichnis `/u00/app/oracle/network/admin` kopiert werden. Alternativ kann man das unten aufgeführte Putty SCP Kommando verwenden.

```
certutil -ca.cert c:\vagrant_common\config\tnsadmin\RootCA_trivadislabs.  
com.cer
```

```
"C:\Program Files\PuTTY\pscp.exe" c:\vagrant_common\config\tnsadmin\  
RootCA_trivadislabs.com.cer ol7db18.trivadislabs.com:/u00/app/oracle/  
network/admin
```

In der Vagrant VM Umgebung ist zudem das Verzeichnis `c:\vagrant_common\` auf allen Systemen verfügbar. Somit lässt sich die Datei einfach auf dem Datenbank Server nutzen.

```
cp /vagrant_common/config/tnsadmin/RootCA_trivadislabs.com.cer /u00/app/  
oracle/network/admin
```

Um Oracle CMU mit Passwort Authentifizierung verwenden zu können, muss Active Directory entsprechend angepasst werden. Dazu muss mit WinSCP die Datei `opwdintg.exe` auf den Active Directory Server kopiert werden. Auf dem Datenbank Server liegt die Datei im Oracle 19c Home `$ORACLE_HOME/bin/opwdintg.exe`. Alternativ man das unten aufgeführte Putty SCP Kommando verwenden.

```
"C:\Program Files\PuTTY\pscp.exe" ol7db18.trivadislabs.com:/u00/app/oracle  
/product/19.0.0.0/bin/opwdintg.exe c:\vagrant_common\config\tnsadmin\  
RootCA_trivadislabs.com.cer
```

In der Vagrant VM Umgebung ist zudem das Verzeichnis `c:\vagrant_common\` auf allen Systemen verfügbar. Somit lässt sich die Datei einfach auf dem Datenbank Server kopieren.

```
ls -alh $ORACLE_HOME/bin/opwdintg.exe  
cp $ORACLE_HOME/bin/opwdintg.exe /vagrant_common/config/tnsadmin
```

Anschliessend muss die Datei auf dem Active Directory ausgeführt werden, um das AD Schema zu erweitern und das Passwort Filter Plugin zu installieren. Dazu wird `opwdintg.exe` direkt in einem Command Shell (`cmd.exe`) ausgeführt.

```
c:\vagrant_common\config\tnsadmin\opwdintg.exe
```

Bei der Installstion sind folgende Fragen mit Ja respektive Yes zu beantworten:

- Do you want to extend AD schema? [Yes/No]:
- Schema extension for this domain will be permanent. Continue? [Yes/No]:
- Do you want to install Oracle password filter?[Yes/No]:
- The change requires machine reboot. Do you want to reboot now?[Yes/No]:

Nachdem der Active Directory Server neu gestartet wurde, müssen zum Abschluss die neu erstellten Gruppen für die Passwort Verifier entsprechend vergeben werden. Entsprechende Benutzer, welche sich an der Datenbank anmelden, müssen dazu ein Oracle Password Hash haben. Dieser wird

vom Password Filter bei allen Benutzer erstellt, welche in der Gruppe *ORA\_VFR\_11G* respektive *ORA\_VFR\_12C* sind. Zudem müssen diese Benutzer ihr Passwort neu setzen, damit das Passwort Filter Plugin auch effektiv das Attribut *orclCommonAttribute* setzt.

**Variante 1:** Anpassen der Gruppe *Trivadislabs Users* und manuelles hinzufügen von *ORA\_VFR\_11G* respektive *ORA\_VFR\_12C* zu MemberOf.

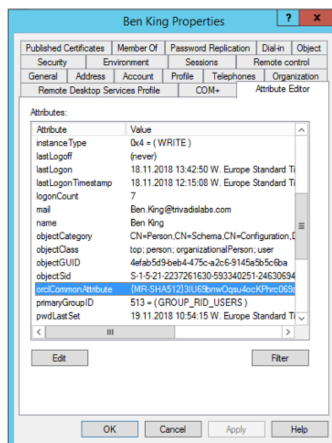
- Starten von *Active Directory Users and Computers*.
- Auswahl der Gruppe *Trivadislabs Users* im Container Groups.
- Öffnen der *Properties* mit rechtem Mausklick.
- Im Tab *Member Of Add...* auswählen.
- Hinzufügen der Gruppe *ORA\_VFR\_11G* respektive *ORA\_VFR\_12C*.
- Schliessen der Dialoge mit *Ok*.

Anschliessend manuelles Anpassen der Passwörter für die gewünschten Benutzer. Dazu muss man in *Active Directory Users and Computers* jeweils den Benutzer auswählen und mit rechtem Mausklick *Reset Password...* starten, um ein neues Passwort zu setzen.

**Variante 2:** Öffnen eines PowerShell Fenster und ausführen des Scripts `c:\aoug\lab\04_cmu\reset_ad_users.ps1`. Das Script passt sowohl die Gruppe an und ändert die Passwörter aller Benutzer.

```
C:\vagrant\scripts\reset_ad_users.ps1
```

Kontrolle ob das Attribut *orclCommonAttribute* gesetzt ist. Die folgende Abbildung zeigt die Properties vom Benutzer King und das Attribut *orclCommonAttribute*.



**Figure 1:** "Benutzereigenschaften Benutzer King"

Benutzer `oracle` für die CMU intergration

## Server und Datenbank Konfiguration

Arbeitsumgebung für diesen Abschnitt:

- **Server:** `ol7db18.trivadislabs.com`
- **Benutzer:** `oracle`
- **Datenbank:** `TDB180S`

CMU benötigt in allen Oracle 18c Versionen einen Patch. Siehe auch Oracle MOS Note [2462012.1](#) *How To Configure Authentication For The Centrally Managed Users In An 18c Database* und Oracle Support Bug OUD 12C: DIGEST-MD5 SASL AUTHENTICATION FAILS IF ORACLECONTEXT ENTRY AND LDAPS [29034231](#). Bevor CMU genutzt werden kann ist zu prüfen ob der Patch installiert ist.

```
$cdh/OPatch/patch lsinventory
```

```
$cdh/OPatch/patch lsinventory |grep -i 28994890
```

Erstellen der SQLNet Konfigurationsdatei `dsi.ora` mit den folgenden Informationen zum Aktive Directory Server. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/aoug/lab` vorhanden.

```
cp $cdl/aoug/lab/dsi.ora $cdn/admin/dsi.ora
```



```
vi $cdn/admin/dsi.ora
DSI_DIRECTORY_SERVERS = (win2016ad.trivadislabs.com:389:636)
DSI_DEFAULT_ADMIN_CONTEXT = "dc=trivadislabs,dc=com"
DSI_DIRECTORY_SERVER_TYPE = AD
```

Erstellen eines neuen Oracle Wallet für die Datenbank TDB180S.

```
mkdir $ORACLE_BASE/admin/$ORACLE_SID/wallet
orapki wallet create -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet -
auto_login
```

Hinzufügen der Einträge für den Benutzername, Passwort und den Distinguished Name.

```
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.USERNAME oracle

mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.DN CN=oracle,CN=Users,DC=trivadislabs,DC=com

mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.PASSWORD LAB01schulung
```

Copy/Paste Variante für die Live Demo

```
echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
createEntry ORACLE.SECURITY.USERNAME oracle

echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
createEntry ORACLE.SECURITY.DN CN=oracle,CN=Users,DC=trivadislabs,DC=
com

echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
createEntry ORACLE.SECURITY.PASSWORD LAB01schulung
```

Laden des Root Zertifikat vom Active Directory Server in das Wallet.

```
orapki wallet add -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet -cert
$TNS_ADMIN/RootCA_trivadislabs.com.cer -trusted_cert
```

Mit folgenden Befehlen lässt sich prüfen, wass nun effektiv im Wallet steht.

```
orapki wallet display -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet

Oracle PKI Tool Release 18.0.0.0.0 - Production
```

```
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights
reserved.

Requested Certificates:
User Certificates:
Oracle Secret Store entries:
ORACLE.SECURITY.DN
ORACLE.SECURITY.PASSWORD
ORACLE.SECURITY.USERNAME
Trusted Certificates:
Subject:          CN=Trivadis LAB Enterprise Root CA,DC=trivadislabs,DC=com
```

Oder der Inhalt vom Wallet mit mkstore

```
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -list
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.DN
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.PASSWORD
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.USERNAME
```

```
echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
list
echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
viewEntry ORACLE.SECURITY.DN
echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
viewEntry ORACLE.SECURITY.PASSWORD
echo LAB01schulung|mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -
viewEntry ORACLE.SECURITY.USERNAME
```

Mit dem LDAP Search Befehl lässt sich zudem Prüfen, ob der Zugriff auf das Active Directory mit dem Wallet funktioniert. Der folgende Befehl sucht nach einem `sAMAccountName=blo`. **Achtung!** Die Passwörter für den Benutzer oracle18c sowie das Wallet Passwort werden hier auf dem Commandline angegeben.

```
ldapsearch -h win2016ad.trivadislabs.com -p 389 \
-D "CN=oracle,CN=Users,DC=trivadislabs,DC=com" \
-w LAB01schulung -U 2 \
-W "file:/u00/app/oracle/admin/$ORACLE_SID/wallet" \
-P LAB01schulung -b "OU=People,DC=trivadislabs,DC=com" \
-s sub "(sAMAccountName=blo*)" dn orclCommonAttribute
```

## Benutzer und Rollen

Arbeitsumgebung für die Übung:

- **Server:** ol7db18.trivadislabs.com
- **Benutzer:** oracle
- **Datenbank:** TDB184A

Als letzter Konfigurationspunkt für Centrally Managed User müssen neben dem Mapping entsprechende init.ora Parameter angepasst werden. Starten Sie ein sqlplus als SYSDBA und setzen Sie die beiden Parameter *ldap\_directory\_access* und *ldap\_directory\_sysauth*.

```
ALTER SYSTEM SET ldap_directory_access = 'PASSWORD';
ALTER SYSTEM SET ldap_directory_sysauth = YES SCOPE=SPFILE;
STARTUP FORCE;
```

Erstellen Sie einen globalen shared Benutzer *tv\_d\_global\_users*, der für alle Mitarbeiter gilt. Also für alle Benutzer in der Gruppe *cn=Trivadis LAB Users,ou=Groups,dc=trivadislabs,dc=com*. Zudem sollen sich alle Benutzer verbinden können.

```
CREATE USER tv_d_global_users IDENTIFIED GLOBALLY AS 'CN=Trivadislabs Users
,OU=Groups,DC=trivadislabs,DC=com';
GRANT create session TO tv_d_global_users ;
GRANT SELECT ON v_$session TO tv_d_global_users ;
```

Verbinden Sie sich als Benutzer Blofeld und prüfen Sie die detail Informationen zu dieser Session wie Authentifizierung, Identity etc.

```
connect "blofeld@TRIVADISLABS.COM"@TDB180S

show user
@sousrinf
```

Nun können sich alle AD Benutzer der Gruppe *Trivadis LAB Users* mit der Datenbank TDB184A verbinden. Sie erhalten dabei die basis Rechte, welche wir zuvor dem Datenbank Account *tv\_d\_global\_users* gegeben haben. Interessant wird es, wenn die Benutzer aus den verschiedenen Abteilungen unterschiedliche Rechte oder Rollen erhalten. Dazu erstellen wir entsprechende Rollen mit einem Mapping auf die Active Directory Gruppe oder Organisation Unit.

```
CONNECT / AS SYSDBA
```

```
CREATE ROLE mgmt_role IDENTIFIED GLOBALLY AS
'CN=Trivadislabs Management,OU=Groups,DC=trivadislabs,DC=com';

CREATE ROLE rd_role IDENTIFIED GLOBALLY AS
'CN=Trivadislabs Developers,OU=Groups,DC=trivadislabs,DC=com';
```

Prüfen Sie nun die verschiedenen Rechte / Rollen der einzelnen Mitarbeitern aus diesen Abteilungen.

```
CONNECT "moneypenny@TRIVADISLABS.COM"/LAB01schulung@TDB180S
SELECT * FROM session_roles;

CONNECT "smith@TRIVADISLABS.COM"/LAB01schulung@TDB180S
SELECT * FROM session_roles;

CONNECT "blofeld@TRIVADISLABS.COM"/LAB01schulung@TDB180S
SELECT * FROM session_roles;
```

### Zusatzaufgabe: Rollen und Administratoren

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an. Erstellen Sie ein global private Schema für den Benutzer Adams. Was für ein DB Benutzer wird jetzt verwendet, wenn sich der Benutzer Adams anmeldet? Welche Rollen sind Aktiv?

```
CONNECT / AS SYSDBA

CREATE USER adams IDENTIFIED GLOBALLY AS 'CN=Douglas Adams,OU=Research,OU=
  People,DC=trivadislabs,DC=com';
GRANT create session TO adams ;
GRANT SELECT ON v_$session TO adams ;

connect "adams@TRIVADISLABS.COM"/LAB01schulung@TDB180S
SELECT * FROM session_roles;
show user
@sousrinf
```

Erstellen Sie ein Mapping für die DBA's, welche sich auch als SYSDBA anmelden sollen. Prüfen Sie dazu als erstes das Format der Oracle Password Datei. Voraussetzung für das Mapping von Administratoren ist die Passwort Datei Version 12.2.

```
orapwd describe file=$cdh/dbs/orapwTDB180S
```

Migrieren Sie die aktuelle Passwort Datei in das Format 12.2. Alternativ können Sie die Passwort Datei auch neu anlegen.

```
mv $cdh/dbs/orapwTDB180S $cdh/dbs/orapwTDB180S_format12
orapwd format=12.2 input_file=$cdh/dbs/orapwTDB180S_format12 file=$cdh/dbs/
/orapwTDB180S
orapwd describe file=$cdh/dbs/orapwTDB180S
```

Erstellen Sie ein Mapping für den DBA Ian Fleming *CN=Ian Fleming,OU=Information Technology,OU=People,DC=trivadislabs,DC=com*

```
CREATE USER fleming IDENTIFIED GLOBALLY AS
'CN=Ian Fleming,OU=Information Technology,OU=People,DC=trivadislabs,DC=com
';
GRANT SYSDBA TO fleming;
GRANT connect TO fleming;
GRANT SELECT ON v_$session TO fleming;
```

Verbinden Sie sich mit und ohne SYSDB als Ian Fleming. Was für Rechte sowie Authentifizierungsinformationen finden Sie?

```
CONNECT "fleming@TRIVADISLABS.COM"/LAB01schulung@TDB180S
SELECT * FROM session_roles;
show user
@sousrinf
```

```
CONNECT "fleming@TRIVADISLABS.COM"/LAB01schulung@ol7db18.trivadislabs.com
:1521/TDB180S as sysdba
SELECT * FROM session_roles;
show user
@sousrinf
```

#Kerberos Authentifizierung

**Übungsziele:** Konfiguration der Kerberos Authentifizierung für die Datenbanken TDB122A und TDB184. Erstellen eines Benutzers mit Kerberos Authentifizierung sowie erfolgreichem Login lokal (Linux VM) und remote (Windows VM).

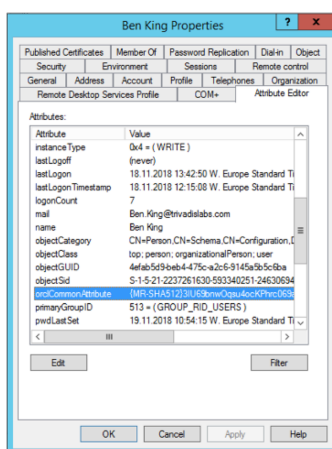
## Service Principle und Keytab Datei

Arbeitsumgebung für die Übung

- **Server:** ad.trivadislabs.com

- **Benutzer:** Administrator

Für die Kerberos Authentifizierung wird ein Service Principle benötigt. Der Entsprechende Benutzer Account wurde vorbereitet. Kontrollieren Sie in auf dem Server *ad.trivadislabs.com* mit dem Tool *Active Directory User and Computers* ob der Benutzer *db.trivadislabs.com* existiert. Falls ja, was hat der Benutzer für Einstellungen bezüglich Login Name und Account optionen? Passen Sie ggf noch die Account Optionen an uns setzen *Kerberos AES 128* und *Kerberos AES 256*. Die folgende Abbildung zeigt ein Beispiel. Optional können Sie den Benutzer auch löschen und neu anlegen.



**Figure 2:** "Benutzereigenschaften"

Nachdem die Account Optionen angepasst wurden, ist für diesen Benutzer eine Keytab Datei zu erstellen. Öffnen Sie dazu ein Command Prompt (*cmd.exe*) und führen *ktpass.exe* aus.

```
ktpass.exe -princ oracle/db.trivadislabs.com@TRIVADISLABS.COM -mapuser db.
trivadislabs.com -pass LAB01schulung -crypto ALL -ptype
KRB5_NT_PRINCIPAL -out C:\u00\app\oracle\network\admin\db.trivadislabs
.com.keytab
```

Überprüfen Sie anschliessend den Service Principle Names (SPN) mit *setspn*

```
setspn -L db.trivadislabs.com
```

Kopieren Sie die Keytab Datei mit WinSCP auf den Datenbank Sever in das Verzeichnis *\$cdn/admin*. Achten Sie darauf, dass die Datei als Binärdatei kopiert wird. Alternativ können Sie auch das unten aufgeführte Putty SCP Kommando verwenden.

```
"C:\Program Files\PuTTY\pscp.exe" C:\u00\app\oracle\network\admin\db.
trivadislabs.com.keytab db.trivadislabs.com:/u00/app/oracle/network/
admin
```

## SQLNet Konfiguration

Arbeitsumgebung für die Übung:

- **Server:** db.trivadislabs.com
- **Benutzer:** oracle

Ergänzen Sie die `sqlnet.ora` Datei mit folgenden Parametern. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/doag2018/lab/03_krb` vorhanden.

```
cp $cdl/doag2018/lab/03_krb/sqlnet.ora $cdn/admin/sqlnet.ora
vi $cdn/admin/sqlnet.ora
#####
# Kerberos Configuration
#####
SQLNET.AUTHENTICATION_SERVICES = (BEQ,KERBEROS5)
SQLNET.FALLBACK_AUTHENTICATION = TRUE
SQLNET.KERBEROS5_KEYTAB = /u00/app/oracle/network/admin/db.trivadislabs.
com.keytab
SQLNET.KERBEROS5_REALMS = /u00/app/oracle/network/admin/krb.realms
SQLNET.KERBEROS5_CC_NAME = /u00/app/oracle/network/admin/krb5.conf
SQLNET.KERBEROS5_CONF = /u00/app/oracle/network/admin/krb5.conf
SQLNET.KERBEROS5_CONF_MIT=TRUE
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE = oracle
```

Erstellen Sie die Kerberos Konfigurationsdatei `krb5.conf` mit folgendem Inhalt. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/doag2018/lab/03_krb` vorhanden.

```
cp $cdl/doag2018/lab/03_krb/krb5.conf $cdn/admin/krb5.conf
vi $cdn/admin/krb5.conf
[libdefaults]
default_realm = TRIVADISLABS.COM
clockskew=300
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true

[realms]
```

```
TRIVADISLABS.COM = {  
    kdc = ad.trivadislabs.com  
    admin_server = ad.trivadislabs.com  
}  
  
[domain_realm]  
.trivadislabs.com = TRIVADISLABS.COM  
trivadislabs.com = TRIVADISLABS.COM
```

Nachdem die Keytab Datei auf dem Datenbank Server kopiert worden ist, kann mit `oklist` überprüft werden was die Datei für Crypto Algorithmen unterstützt. Somit wird zudem indirekt geprüft ob die Keytab Datei verwendet werden kann.

```
oklist -e -k $cdn/admin/ol7db18.trivadislabs.com.keytab
```

Kontrollieren Sie ob die Namensauflösung wie gewünscht funktioniert.

```
nslookup win2016ad.trivadislabs.com  
nslookup 10.0.0.4  
nslookup ol7db18.trivadislabs.com  
nslookup 10.0.0.5
```

Erstellen Sie anschliessend mit `okinit` manuell ein Session Ticket.

```
okinit king@TRIVADISLABS.COM  
  
oklist
```

## Kerberos Authentifizierung

Arbeitsumgebung für die Übung:

- **Server:** ol7db18.trivadislabs.com
- **Benutzer:** oracle

Passen Sie den `init.ora` Parameter `OS Prefix` an. Für die Kerberos Authentifizierung muss dieser leer sein.

```
sql  
Show parameter os_authent_prefix  
ALTER SYSTEM SET os_authent_prefix='' SCOPE=spfile;  
STARTUP FORCE;
```



Erstellen Sie einen Kerberos Benutzers für den Mitarbeiter King. Verwenden Sie dazu die Variante mit dem Kerberos Principal Name.

```
CREATE USER king IDENTIFIED EXTERNALLY AS 'king@TRIVADISLABS.COM';
GRANT CONNECT TO king;
GRANT SELECT ON v_$session TO king;
```

Login als Benutzer King mit dem zuvor generierten Session Ticket und anzeigen der Informationen zur aktuellen Session.

```
connect /@TDB180S

show user

@sousrinf

SELECT sys_context('USERENV','AUTHENTICATION_TYPE') FROM DUAL;
SELECT sys_context('USERENV','AUTHENTICATION_METHOD') FROM DUAL;
SELECT sys_context('USERENV','AUTHENTICATED_IDENTITY') FROM DUAL;
```

## Zusatzaufgaben

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an:

- Versuchen Sie einen weiteren Kerberos Benutzer in der Datenbank zu erstellen (z.B. *adams*). Dabei nutzen Sie aber den UPN als Benutzernamen. Wie setzt sich dieser genau zusammen? Wie wird dieser bei einem `CREATE USER` geschrieben, damit die Authentifizierung klappt?
- Kombinieren Sie Kerberos Authentifizierung mit Proxy Authentifizierung.
- Locken Sie im AD den Benutzer und versuchen erneut mit der Datenbank zu verbinden.
- Versuchen Sie sich als Benutzer *king* am Active directory Server anzumelden und eine SQLPlus Verbindung auf den Datenbank Server zu öffnen. Hierzu müssen Sie vorgängig noch die Kerberos Client Konfiguration erstellen (sqlnet.ora sowie die krb5.conf Datei erstellen).

Wie ist das jetzt mit Kerberos? Wenn Sie die Übung zu Kerberos erfolgreich abgeschlossen haben, können sich die Benutzer nun auch mit Kerberos Authentifizieren. Ein Versuch mit dem Benutzer *Bond* schafft hier Klarheit. Generieren sie zuerst manuell ein Ticket Granting Ticket mit `okinit`. Die Passwortabfrage umgehen wir bei diesem Beispiel einfach indem wir das Passwort mit einem `echo` | via STDIN an `okinit` schicken. Mit dem Skript `sousrinf.sql` sehen wir anschliessend detaillierte Informationen zur Authentifizierung.

```
sqh
```

```
host echo LAB01schulung|okinit bond
connect /@TDB180S
SELECT * FROM session_roles;

show user
@sousrinf.sql
```

```
sqh
host echo LAB01schulung|okinit fleming
connect /@TDB180S
SELECT * FROM session_roles;

show user
@sousrinf.sql

connect /@TDB180S as sysdba
```

```
echo LAB01schulung|okinit bond
sqlplus /@TDB184A
SELECT * FROM session_roles;

show user
@sousrinf.sql
```

## Übungen: Datenbank Authentifizierung und Password Verifier

**Übungsziele:** Kennenlernen der Übungsumgebung, BasEnv sowie der Datenbanken. Festigen der Kenntnisse im Bereich Password Authentifizierung und Password Hashes.

Arbeitsumgebung für die Übung

- **Server:** db.trivadislabs.com
- **DB:** TDB122A oder TDB184A

Die folgenden Aufgaben und Beispiele werden auf der DB TDB184A durchgeführt. Grundsätzlich können diese aber auch auf TDB122A ausgeführt werden.

### Überprüfung der aktuellen Password Verifier

1. Prüfen Sie was aktuell für Password Hashes in der Datenbank vorhanden sind. Welche Hashes gibt es? Wieso sind bei gewissen Benutzer keine Angaben in *password\_versions*?

```
set linesize 120 pagesize 200
col USERNAME for a25
SELECT username, password_versions FROM dba_users;
```

- Prüfen Sie wie die VIEW *dba\_users* auf die Information zu *password\_versions* kommt. Im Code zum View *dba\_users* findet man entsprechende *decode* Funktionen wo auf die Spalten *u.password* und *u.spare4* zugegriffen wird.

```
set linesize 120 pagesize 200
set long 200000
SELECT text FROM dba_views WHERE view_name='DBA_USERS';
```

- Was für Passwort Hashes hat der Benutzer SCOTT effektiv?

```
set linesize 120 pagesize 200
col password for a16
col spare4 for a40
SELECT password, spare4 FROM user$ WHERE name='SCOTT';
```

- Kontrollieren Sie was in der Datei *sqlnet.ora* für die Parameter *\*ALLOWED\_LOGON\_VERSION\_\** definiert wurde. Verwenden Sie alternative *cat*, *less*, *more* oder *vi* um den Inhalt von *sqlnet.ora* anzuzeigen.

```
less $cdn/admin/sqlnet.ora

cat $cdn/admin/sqlnet.ora|grep -i ALLOWED_LOGON_VERSION
```

- Prüfen Sie was von SQLNet effektiv verwendet wird.

- Einschalten des SQLNet Tracing auf der Client Seite. Setzen von *DIAG\_ADR\_ENABLED* und *TRACE\_LEVEL\_CLIENT*. Anbei manuell mit *vi* oder alternativ direkt mit *sed* ersetzen lassen.

```
vi $cdn/admin/sqlnet.ora
DIAG_ADR_ENABLED=OFF
TRACE_LEVEL_CLIENT=SUPPORT
```

```
sed -i "s|DIAG_ADR_ENABLED.*|DIAG_ADR_ENABLED=OFF|" $cdn/admin/sqlnet.ora
sed -i "s|TRACE_LEVEL_CLIENT.*|TRACE_LEVEL_CLIENT=SUPPORT|" $cdn/admin/
sqlnet.ora
```

- Löschen Sie allfällige alte Trace Dateien.

```
rm $cdn/trc/sqlnet_client_*.trc
```

- Verbinden als Benutzer Scott

```
sqlplus scott/tiger
```

```
show user
```

- Kontrollieren Sie die Trace Datei. Was ist für ALLOWED\_LOGON\_VERSION gesetzt? Falls nichts gesetzt ist, was für ein Wert gilt?

```
ls -rtl $cdn/trc
```

```
less $cdn/trc/sqlnet_client_*.trc
```

```
grep -i ALLOWED_LOGON_VERSION $cdn/trc/sqlnet_client_*.trc
```

- Schalten Sie das Tracing wieder aus.

```
vi $cdn/admin/sqlnet.ora
```

```
DIAG_ADR_ENABLED=ON
```

```
TRACE_LEVEL_CLIENT=OFF
```

```
sed -i "s|DIAG_ADR_ENABLED.*|DIAG_ADR_ENABLED=ON|" $cdn/admin/sqlnet.ora
```

```
sed -i "s|TRACE_LEVEL_CLIENT.*|TRACE_LEVEL_CLIENT=OFF|" $cdn/admin/sqlnet.  
ora
```

## Anpassen der Password Verifier

1. Löschen Sie den Oracle 12c Password Hash vom Benutzer Scott. Respektive setzen Sie explizit den 11g Hashes.

```
SELECT spare4 FROM user$ WHERE name='SCOTT';
```

```
set linesize 170
```

```
col 11G_HASH for a62
```

```
SELECT
```

```
    REGEXP_SUBSTR(spare4,'(S:[[:alnum:]]+)') "11G_HASH"
```

```
FROM user$ WHERE name='SCOTT';
```

```
col 12C_HASH for a162
```

```
SELECT
  REGEXP_SUBSTR(spare4,'(T:[[:alnum:]]+)') "12C_HASH"
FROM user$ WHERE name='SCOTT';

ALTER USER scott IDENTIFIED BY VALUES 'S:54
A0B23AE639D4E0E22963A65A380DD496B8FCB65D1A5F9CC910EE625D8C';
```

2. Kontrolle der *password\_versions* vom Benutzer *SCOTT*.

```
col username for a30
SELECT username,password_versions FROM dba_users WHERE username='SCOTT';
```

3. Anpassen des SQLNet Parameter *ALLOWED\_LOGON\_VERSION\_SERVER* und setzen des 12a Authentifizierungsprotokolls.

```
sed -i "s|#SQLNET.ALLOWED_LOGON_VERSION_SERVER.*|SQLNET.
ALLOWED_LOGON_VERSION_SERVER=12a|" $cdn/admin/sqlnet.ora
sed -i "s|#SQLNET.ALLOWED_LOGON_VERSION_SERVER.*|SQLNET.
ALLOWED_LOGON_VERSION_SERVER=12a|" $cdn/admin/sqlnet.ora
```

4. Als User Scott verbinden. Kann man sich überhaupt verbinden?

```
sqlplus scott/tiger

show user
```

1. Anpassen des SQLNet Parameter *ALLOWED\_LOGON\_VERSION\_CLIENT* und *ALLOWED\_LOGON\_VERSION\_SERVER*, setzen des 11 Authentifizierungsprotokolls.

```
sed -i "s|#ALLOWED_LOGON_VERSION_CLIENT.*|SQLNET.
ALLOWED_LOGON_VERSION_CLIENT=11|" $cdn/admin/sqlnet.ora
sed -i "s|#SQLNET.ALLOWED_LOGON_VERSION_CLIENT.*|SQLNET.
ALLOWED_LOGON_VERSION_CLIENT=11|" $cdn/admin/sqlnet.ora
sed -i "s|#SQLNET.ALLOWED_LOGON_VERSION_SERVER.*|SQLNET.
ALLOWED_LOGON_VERSION_SERVER=11|" $cdn/admin/sqlnet.ora
sed -i "s|#SQLNET.ALLOWED_LOGON_VERSION_SERVER.*|SQLNET.
ALLOWED_LOGON_VERSION_SERVER=11|" $cdn/admin/sqlnet.ora
```

6. Als User Scott verbinden. Kann man sich überhaupt verbinden?

```
sqlplus scott/tiger

show user
```

7. Was passiert wenn man als *SYS* das Passwort von *SCOTT* neu setzt? Welcher Passwort Hash hat *SCOTT* nun?

```
connect / as sysdba

ALTER USER scott IDENTIFIED BY tiger;

set linesize 120 pagesize 200
col USERNAME for a25
col password for a16
col spare4 for a40
SELECT username, password_versions FROM dba_users WHERE username='SCOTT';
SELECT password, spare4 FROM user$ WHERE name='SCOTT';
```

## Zusatzaufgaben

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an:

- Setzen von *ALLOWED\_LOGON\_VERSION\_CLIENT* und *ALLOWED\_LOGON\_VERSION\_SERVER* auf Werte kleiner 11 z.B 10, 9 oder 8. Was bekommt der Benutzer *SCOTT* für Passwort Hashes wenn man als *SYS* das Passwort mit *ALTER USER* neu setzt?
- Welches Passwort wird beim Login verwendet?

## Übungen: Kerberos Authentifizierung

**Übungsziele:** Konfiguration der Kerberos Authentifizierung für die Datenbanken TDB122A und TDB184. Erstellen eines Benutzers mit Kerberos Authentifizierung sowie erfolgreichem Login lokal (Linux VM) und remote (Windows VM).

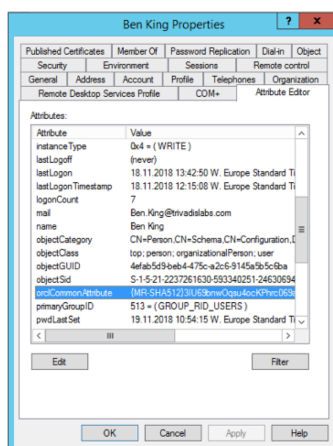
### Service Principle und Keytab Datei

Arbeitsumgebung für die Übung

- **Server:** ad.trivadislabs.com
- **Benutzer:** Administrator

Für die Kerberos Authentifizierung wird ein Service Principle benötigt. Der Entsprechende Benutzer Account wurde vorbereitet. Kontrollieren Sie in auf dem Server *ad.trivadislabs.com* mit dem Tool *Active Directory User and Computers* ob der Benutzer *db.trivadislabs.com* existiert. Falls ja, was hat der Benutzer für Einstellungen bezüglich Login Name und Account optionen? Passen Sie ggf

noch die Account Optionen an uns setzen *Kerberos AES 128* und *Kerberos AES 256*. Die folgende Abbildung zeigt ein Beispiel. Optional können Sie den Benutzer auch löschen und neu anlegen.



**Figure 3:** "Benutzereigenschaften"

Nachdem die Account Optionen angepasst wurden, ist für diesen Benutzer eine Keytab Datei zu erstellen. Öffnen Sie dazu ein Command Prompt (`cmd.exe`) und führen `ktpass.exe` aus.

```
ktpass.exe -princ oracle/db.trivadislabs.com@TRIVADISLABS.COM -mapuser db.
trivadislabs.com -pass LAB01schulung -crypto ALL -ptype
KRB5_NT_PRINCIPAL -out C:\u00\app\oracle\network\admin\db.trivadislabs
.com.keytab
```

Überprüfen Sie anschliessend den Service Principle Names (SPN) mit `setspn`

```
setspn -L db.trivadislabs.com
```

Kopieren Sie die Keytab Datei mit WinSCP auf den Datenbank Sever in das Verzeichnis `$cdn/admin`. Achten Sie darauf, dass die Datei als Binärdatei kopiert wird. Alternativ können Sie auch das unten aufgeführte Putty SCP Kommando verwenden.

```
"C:\Program Files\PuTTY\pscp.exe" C:\u00\app\oracle\network\admin\db.
trivadislabs.com.keytab db.trivadislabs.com:/u00/app/oracle/network/
admin
```

## SQLNet Konfiguration

Arbeitsumgebung für die Übung:

- **Server:** db.trivadislabs.com
- **Benutzer:** oracle

Ergänzen Sie die `sqlnet.ora` Datei mit folgenden Parametern. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/doag2018/lab/03_krb` vorhanden.

```
cp $cdl/doag2018/lab/03_krb/sqlnet.ora $cdn/admin/sqlnet.ora
vi $cdn/admin/sqlnet.ora
#####
# Kerberos Configuration
#####
SQLNET.AUTHENTICATION_SERVICES = (BEQ,KERBEROS5)
SQLNET.FALLBACK_AUTHENTICATION = TRUE
SQLNET.KERBEROS5_KEYTAB = /u00/app/oracle/network/admin/db.trivadislabs.
                        com.keytab
SQLNET.KERBEROS5_REALMS = /u00/app/oracle/network/admin/krb.realms
SQLNET.KERBEROS5_CC_NAME = /u00/app/oracle/network/admin/krb5.conf
SQLNET.KERBEROS5_CONF = /u00/app/oracle/network/admin/krb5.conf
SQLNET.KERBEROS5_CONF_MIT=TRUE
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE = oracle
```

Erstellen Sie die Kerberos Konfigurationsdatei `krb5.conf` mit folgendem Inhalt. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/doag2018/lab/03_krb` vorhanden.

```
cp $cdl/doag2018/lab/03_krb/krb5.conf $cdn/admin/krb5.conf
vi $cdn/admin/krb5.conf
[libdefaults]
    default_realm = TRIVADISLABS.COM
    clockskew=300
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true

[realms]
    TRIVADISLABS.COM = {
        kdc = ad.trivadislabs.com
        admin_server = ad.trivadislabs.com
    }

[domain_realm]
```



```
.trivadislabs.com = TRIVADISLABS.COM  
trivadislabs.com = TRIVADISLABS.COM
```

Nachdem die Keytab Datei auf dem Datenbank Server kopiert worden ist, kann mit `oklist` überprüft werden was die Datei für Crypto Algorithmen unterstützt. Somit wird zudem indirekt geprüft ob die Keytab Datei verwendet werden kann.

```
oklist -e -k $cdn/admin/db.trivadislabs.com.keytab
```

Kontrollieren Sie ob die Namensauflösung wie gewünscht funktioniert.

```
nslookup ad.trivadislabs.com  
nslookup 10.0.0.4  
nslookup db.trivadislabs.com  
nslookup 10.0.0.5
```

Erstellen Sie anschliessend mit `okinit` manuell ein Session Ticket.

```
okinit king@TRIVADISLABS.COM  
  
oklist
```

## Kerberos Authentifizierung

Arbeitsumgebung für die Übung:

- **Server:** db.trivadislabs.com
- **Benutzer:** oracle

Passen Sie den init.ora Parameter OS Prefix an. Für die Kerberos Authentifizierung muss dieser leer sein.

```
sql  
Show parameter os_authent_prefix  
ALTER SYSTEM SET os_authent_prefix='' SCOPE=spfile;  
STARTUP FORCE;
```

Erstellen Sie einen Kerberos Benutzers für den Mitarbeiter King. Verwenden Sie dazu die Variante mit dem Kerberos Principal Name.

```
CREATE USER king IDENTIFIED EXTERNALLY AS 'king@TRIVADISLABS.COM';  
GRANT CONNECT TO king;  
GRANT SELECT ON v_$session TO king;
```

Login als Benutzer King mit dem zuvor generierten Session Ticket und anzeigen der Informationen zur aktuellen Session.

```
connect /@TDB184A

show user

@sousrinf

SELECT sys_context('USERENV','AUTHENTICATION_TYPE') FROM DUAL;
SELECT sys_context('USERENV','AUTHENTICATION_METHOD') FROM DUAL;
SELECT sys_context('USERENV','AUTHENTICATED_IDENTITY') FROM DUAL;
```

## Zusatzaufgaben

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an:

- Versuchen Sie einen weiteren Kerberos Benutzer in der Datenbank zu erstellen (z.B. *adams*). Dabei nutzen Sie aber den UPN als Benutzernamen. Wie setzt sich dieser genau zusammen? Wie wird dieser bei einem `CREATE USER` geschrieben, damit die Authentifizierung klappt?
- Kombinieren Sie Kerberos Authentifizierung mit Proxy Authentifizierung.
- Locken Sie im AD den Benutzer und versuchen erneut mit der Datenbank zu verbinden.
- Versuchen Sie sich als Benutzer *king* am Active directory Server anzumelden und eine SQLPlus Verbindung auf den Datenbank Server zu öffnen. Hierzu müssen Sie vorgängig noch die Kerberos Client Konfiguration erstellen (sqlnet.ora sowie die krb5.conf Datei erstellen).

## Übungen: Centrally Managed User 18c

**Übungsziele:** Konfiguration von Centrally Managed Users für die Datenbank TDB184. Erweitern des Active Directory Schemas inklusive der Installation des Password Filter Plugins. Erstellen von Mappings für Benutzer und Rollen sowie erfolgreichem Login mit Passwort sowie Kerberos Authentifizierung.

### Active Directory

Arbeitsumgebung für die Übung:

- **Server:** ad.trivadislabs.com
- **Benutzer:** Administrator

Die folgenden Arbeiten werden in der Regel in Zusammenarbeit mit dem Windows respektive Active Directory Administrator durchgeführt. Je nach Unternehmensgrösse sind allenfalls noch weitere IT Bereiche mit involviert.

Für das Oracle Wallet wird das Root Zertifikat vom Active Directory Server benötigt. Dieses kann in der Übungsumgebung einfach via Commandline exportiert werden. Öffnen Sie dazu ein Command Prompt (`cmd.exe`) und exportieren das Root Zertifikat. Das exportierte Root Zertifikat müssen Sie anschliessend mit WinSCP auf den Datenbank Server kopieren in das Verzeichnis `/u00/app/oracle/network/admin` kopieren. Alternativ können Sie auch das unten aufgeführte Putty SCP Kommando verwenden.

```
certutil -ca.cert c:\u00\app\oracle\network\admin\Trivadis_LAB_root.cer  
  
"C:\Program Files\PuTTY\pscp.exe" c:\u00\app\oracle\network\admin\  
Trivadis_LAB_root.cer db.trivadislabs.com:/u00/app/oracle/network/admin
```

Um Oracle CMU mit Passwort Authentifizierung verwenden zu können, muss Active Directory entsprechend angepasst werden. Dazu wird mit WinSCP die Datei `opwdintg.exe` auf den Active Directory Server kopiert werden. Auf dem Datenbank Server liegt die Datei im Oracle 18c Home `$ORACLE_HOME/bin/opwdintg.exe`. Alternativ können Sie auch das unten aufgeführte Putty SCP Kommando verwenden.

```
"C:\Program Files\PuTTY\pscp.exe" db.trivadislabs.com:/u00/app/oracle/  
product/18.4.0.0/bin/opwdintg.exe c:\u00\app\oracle\network\admin\
```

Anschliessend muss die Datei auf dem Active Directory ausgeführt werden, um das AD Schema zu erweitern und das Passwort Filter Plugin zu installieren. Öffnen Sie dazu ein Command Prompt (`cmd.exe`) und führen `opwdintg.exe` aus.

```
c:\u00\app\oracle\network\admin\opwdintg.exe
```

Bei der Installation sind folgende Fragen mit Ja respektive Yes zu beantworten:

- Do you want to extend AD schema? [Yes/No]:
- Schema extension for this domain will be permanent. Continue? [Yes/No]:
- Do you want to install Oracle password filter?[Yes/No]:
- The change requires machine reboot. Do you want to reboot now?[Yes/No]:

Nachdem der Active Directory Server neu gestartet wurde, müssen zum Abschluss die neu erstellten Gruppen für die Passwort Verifier entsprechend vergeben werden. Entsprechende Benutzer, welche sich an der Datenbank anmelden, müssen dazu ein Oracle Password Hash haben. Dieser wird vom Password Filter bei allen Benutzer erstellt, welche in der Gruppe `ORA_VFR_11G` respektive



## Server und Datenbank Konfiguration

Arbeitsumgebung für die Übung:

- **Server:** db.trivadislabs.com
- **Benutzer:** oracle
- **Datenbank:** TDB184A

Erstellen Sie ein SQLNet Konfigurationsdatei `dsi.ora` mit den folgenden Informationen zum Active Directory Server. Eine Beispiel Konfigurationsdatei ist im Verzeichnis `$cdl/doag2018/lab/04_cmu` vorhanden.

```
cp $cdl/doag2018/lab/04_cmu/dsi.ora $cdn/admin/dsi.ora
vi $cdn/admin/dsi.ora
DSI_DIRECTORY_SERVERS = (ad.trivadislabs.com:389:636)
DSI_DEFAULT_ADMIN_CONTEXT = "dc=trivadislabs,dc=com"
DSI_DIRECTORY_SERVER_TYPE = AD
```

Erstellen Sie ein neues Oracle Wallet für die Datenbank TDB184A.

```
mkdir $ORACLE_BASE/admin/$ORACLE_SID/wallet
orapki wallet create -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet -
auto_login
```

Fügen Sie die Einträge für den Benutzername, Passwort und den Distinguished Name hinzu.

```
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.USERNAME oracle18c

mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.DN CN=oracle18c,CN=Users,DC=trivadislabs,DC=com

mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -createEntry ORACLE.
SECURITY.PASSWORD LAB01schulung
```

Laden Sie abschliessend noch das Root Zertifikat vom Active Directory Server in das Wallet

```
orapki wallet add -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet -cert
$TNS_ADMIN/Trivadis_LAB_root.cer -trusted_cert
```

Mit folgenden Befehlen lässt sich prüfen, wass nun effektiv im Wallet steht.

```
orapki wallet display -wallet $ORACLE_BASE/admin/$ORACLE_SID/wallet
```

```
Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights
reserved.

Requested Certificates:
User Certificates:
Oracle Secret Store entries:
ORACLE.SECURITY.DN
ORACLE.SECURITY.PASSWORD
ORACLE.SECURITY.USERNAME
Trusted Certificates:
Subject:          CN=Trivadis LAB Enterprise Root CA,DC=trivadislabs,DC=com
```

```
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -list
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.DN
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.PASSWORD
mkstore -wrl $ORACLE_BASE/admin/$ORACLE_SID/wallet -viewEntry ORACLE.
SECURITY.USERNAME
```

Mit dem LDAP Search Befehl lässt sich zudem Prüfen, ob der Zugriff auf das Active Directory mit dem Wallet funktioniert. Der folgende Befehl sucht nach einem `sAMAccountName=blo`. **Achtung!** Die Passwörter für den Benutzer `oracle18c` sowie das Wallet Passwort werden hier auf dem Commandline angegeben.

```
ldapsearch -h ad.trivadislabs.com -p 389 \
-D "CN=oracle18c,CN=Users,DC=trivadislabs,DC=com" \
-w LAB01schulung -U 2 \
-W "file:/u00/app/oracle/admin/TDB184A/wallet" \
-P LAB01schulung -b "OU=People,DC=trivadislabs,DC=com" \
-s sub "(sAMAccountName=blo*)" dn orclCommonAttribute
```

## Benutzer und Rollen

Arbeitsumgebung für die Übung:

- **Server:** db.trivadislabs.com
- **Benutzer:** oracle
- **Datenbank:** TDB184A

Als letzter Konfigurationspunkt für Centrally Managed User müssen neben dem Mapping entsprechende init.ora Parameter angepasst werden. Starten Sie ein sqlplus als SYSDBA und setzen Sie die beiden Parameter *ldap\_directory\_access* und *ldap\_directory\_sysauth*.

```
ALTER SYSTEM SET ldap_directory_access = 'PASSWORD';
ALTER SYSTEM SET ldap_directory_sysauth = YES SCOPE=SPFILE;
STARTUP FORCE;
```

Erstellen Sie einen globalen shared Benutzer *tvd\_global\_users*, der für alle Mitarbeiter gilt. Also für alle Benutzer in der Gruppe *cn=Trivadis LAB Users,ou=Groups,dc=trivadislabs,dc=com*. Zudem sollen sich alle Benutzer verbinden können.

```
CREATE USER tvd_global_users IDENTIFIED GLOBALLY AS 'cn=Trivadis LAB Users
,ou=Groups,dc=trivadislabs,dc=com';
GRANT create session TO tvd_global_users ;
GRANT SELECT ON v_$session TO tvd_global_users ;
```

Verbinden Sie sich als Benutzer Blofeld und prüfen Sie die detail Informationen zu dieser Session wie Authentifizierung, Identity etc.

```
connect "blofeld@TRIVADISLABS.COM"@TDB184A

show user
@sousrinf
```

Nun können sich alle AD Benutzer der Gruppe *Trivadis LAB Users* mit der Datenbank TDB184A verbinden. Sie erhalten dabei die basis Rechte, welche wir zuvor dem Datenbank Account *tvd\_global\_users* gegeben haben. Interessant wird es, wenn die Benutzer aus den verschiedenen Abteilungen unterschiedliche Rechte oder Rollen erhalten. Dazu erstellen wir entsprechende Rollen mit einem Mapping auf die Active Directory Gruppe oder Organisation Unit.

```
CONNECT / AS SYSDBA

CREATE ROLE mgmt_role IDENTIFIED GLOBALLY AS
'CN=Trivadis LAB Management,OU=Groups,DC=trivadislabs,DC=com';

CREATE ROLE rd_role IDENTIFIED GLOBALLY AS
'CN=Trivadis LAB Developers,OU=Groups,DC=trivadislabs,DC=com';
```

Prüfen Sie nun die verschiedenen Rechte / Rollen der einzelnen Mitarbeitern aus diesem Abteilungen.

```
CONNECT "moneypenny@TRIVADISLABS.COM"/LAB01schulung@TDB184A
SELECT * FROM session_roles;
```

```
CONNECT "smith@TRIVADISLABS.COM"/LAB01schulung@TDB184A
SELECT * FROM session_roles;

CONNECT "blofeld@TRIVADISLABS.COM"/LAB01schulung@TDB184A
SELECT * FROM session_roles;
```

Wie ist das jetzt mit Kerberos? Wenn Sie die Übung zu Kerberos erfolgreich abgeschlossen haben, können sich die Benutzer nun auch mit Kerberos Authentifizieren. Ein Versuch mit dem Benutzer *Bond* schafft hier Klarheit. Generieren Sie zuerst manuell ein Ticket Granting Ticket mit `okinit`. Die Passwortabfrage umgehen wir bei diesem Beispiel einfach indem wir das Passwort mit einem `echo` | via STDIN an `okinit` schicken. Mit dem Skript `sousrinf.sql` sehen wir anschliessend detaillierte Informationen zur Authentifizierung.

```
sqh
host echo LAB01schulung|okinit bond
connect /@TDB184A
SELECT * FROM session_roles;

show user
@sousrinf.sql
```

```
echo LAB01schulung|okinit bond
sqlplus /@TDB184A
SELECT * FROM session_roles;

show user
@sousrinf.sql
```

## Zusatzaufgabe: Rollen und Administratoren

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an. Erstellen Sie ein global private Schema für den Benutzer Adams. Was für ein DB Benutzer wird jetzt verwendet, wenn sich der Benutzer Adams anmeldet? Welche Rollen sind Aktiv?

```
CREATE USER adams IDENTIFIED GLOBALLY AS 'CN=Douglas Adams,OU=Research,OU=
  People,DC=trivadislabs,DC=com';
GRANT create session TO adams ;
GRANT SELECT ON v_$session TO adams ;

sqlplus "adams@TRIVADISLABS.COM"/LAB01schulung@TDB184A
```



```
SELECT * FROM session_roles;
show user
@sousrinf
```

Erstellen Sie ein Mapping für die DBA's, welche sich auch als SYSDBA anmelden sollen. Prüfen Sie dazu als erstes das Format der Oracle Password Datei. Voraussetzung für das Mapping von Administratoren ist die Passwort Datei Version 12.2.

```
orapwd describe file=$cdh/dbs/orapwTDB184A
```

Migrieren Sie die aktuelle Passwort Datei in das Format 12.2. Alternativ können Sie die Passwort Datei auch neu anlegen.

```
mv $cdh/dbs/orapwTDB184A $cdh/dbs/orapwTDB184A_format12
orapwd format=12.2 input_file=$cdh/dbs/orapwTDB184A_format12 file=$cdh/dbs
/orapwTDB184A
orapwd describe file=$cdh/dbs/orapwTDB184A
```

Erstellen Sie ein Mapping für den DBA Ian Fleming *CN=Ian Fleming,OU=Information Technology,OU=People,DC=trivadislabs,DC=com*

```
CREATE USER fleming IDENTIFIED GLOBALLY AS
'CN=Ian Fleming,OU=Information Technology,OU=People,DC=trivadislabs,DC=com
';
GRANT SYSDBA TO fleming;
GRANT connect TO fleming;
GRANT SELECT ON v_$session TO fleming;
```

Verbinden Sie sich mit und ohne SYSDB als Ian Fleming. Was für Rechte sowie Authentifizierungsinformationen finden Sie?

```
CONNECT "fleming@TRIVADISLABS.COM"/LAB01schulung@TDB184A
SELECT * FROM session_roles;
show user
@sousrinf
```

```
CONNECT "fleming@TRIVADISLABS.COM"/LAB01schulung@db.trivadislabs.com:1521/
TDB184A as sysdba
SELECT * FROM session_roles;
show user
@sousrinf
```

Versuchen Sie ein weiteres Mapping auf einen anderen global shared Datenbankbenutzer zu machen. Funktioniert das? Was gibt dies für Probleme?

## Übungen: Oracle Unified Directory

**Übungsziele:** Erstellen OUD Directory Server Instanz sowie einer OUD Proxy Server Instanz für die Active Directory Integration. Die Proxy Server Instanz wird im Folgenden für die Übungen mit Enterprise User Security benötigt.

Arbeitsumgebung für die Übung:

- **Server:** oud.trivadislabs.com
- **Benutzer:** oracle

### Oracle Unified Directory Server Instanz

Erstellen Sie eine OUD Directory Server Instanz mit `oud-setup`. Wenn ein X11 Client vorhanden ist, wird das Tool im GUI Mode gestartet. Falls kein X11 Client wird automatisch in den Character mode gewechselt.

```
export ORACLE_HOME=/u00/app/oracle/product/oud12.2.1.3.0
cd $ORACLE_HOME/oud
./oud-setup
```

Verwenden Sie für die OUD Instanz folgende Angaben:

- Instance Path : **/u01/instances/oud\_doag/ODU**
- Do you want to enable the LDAP administration port? Note that some of the OUD tools require this port to be enabled (yes / no) [yes]: **yes**
- On which port would you like the LDAP Administration Connector to accept connections? [4444]: **5444**
- Do you want to enable the HTTP administration port? (yes / no) [no]: **no**
- What would you like to use as the initial root user DN for the Directory Server? [cn=Directory Manager]: **cn=Directory Manager**
- Please provide the password to use for the initial root user: **LAB01schulung**
- Do you want to enable LDAP? (yes / no) [yes]: **yes**
- On which port would you like the Directory Server to accept connections from LDAP clients? [1389]: **2389**
- Do you want to enable Start TLS on LDAP Port "2389"? (yes / no) [no]: **yes**
- Do you want to enable HTTP? (yes / no) [no]: **no**

- Do you want to enable LDAPS? (yes / no) [no]: **yes**
- On which port would you like the Directory Server to accept connections from LDAPS clients? [1636]: **2636**
- Do you want to enable HTTPS? (yes / no) [no]: **no**
- Select Certificate server options: **1**
- Provide the fully-qualified host name or IP address that will be used to generate the self-signed certificate [oud.trivadislabs.com]: **oud.trivadislabs.com**
- Do you want to create base DN's in the server? (yes / no) [yes]: **yes**
- Provide the base DN for the directory data: [dc=example,dc=com]: **dc=trivadislabs,dc=com**
- Options for populating the database: **1**
- Specify the Oracle components with which the server integrates: **1** EUS wird später als Zusatzaufgabe manuell konfiguriert.
- How do you want the OUD server to be tuned? **2**
- How do you want the off-line tools (import-ldif, export-ldif, verify-index and rebuild-index) to be tuned? **3**
- Do you want to start the server when the configuration is completed? (yes / no) [yes]: **yes**
- What would you like to do? **1, 2** oder **3** Wobei dann effektiv 1 gewählt werden muss um die Instanz anzulegen.

Erstellen Sie einen Eintrag in der oudtab Datei. Diese wird für das Setzen der Umgebung mit OUD base benötigt.

```
echo "oud_doag:2389:2636:5444::OUD:N" >> ${ETC_BASE}/oudtab
mkdir -p /u01/admin/oud_doag/etc/
echo "LAB01schulung" >/u01/admin/oud_doag/etc/oud_doag_pwd.txt
```

Laden Sie die Umgebung für die Instanz oud\_doag neu.

```
. oudenv.sh oud_doag
```

Wir haben nun eine simple OUD Directory Server Instanz erstellt. Die Instanz enthält aktuell nichts weiteres als den Basis Eintrag *dc=trivadislabs,dc=com*. Im Rahmen der Zusatzaufgaben wird diese Instanz weiter genutzt.

## Oracle Unified Directory Proxy Instanz

Für die Konfiguration der Proxy besteht die Möglichkeit den GUI Mode zu verwenden oder direkt `oud-proxy-setup` mit cli Parameter. Im folgenden werden wir die OUD Proxy Instanz schrittweise mit Scripten konfigurieren.

1. Vorbereiten der Umgebung. Erstellen eines OUDTAB Eintrages und laden der Umgebung.

```
echo "oud_adproxy:1389:1636:4444::OUD:Y" >> ${ETC_BASE}/oudtab
mkdir -p /u01/admin/oud_adproxy/etc
echo "LAB01schulung" >/u01/admin/oud_adproxy/etc/oud_adproxy_pwd.txt
. oudenv.sh oud_adproxy
```

## 2. Kopieren der Template Scripte von OUD Base

```
cp $cdl/doag2018/lab/05_oud/??_* /u01/admin/oud_adproxy/create
```

## 3. Anpassen respektive Prüfen der Parameter in 00\_init\_environment.

```
vi /u01/admin/oud_adproxy/create/00_init_environment
```

## 4. Erstellen der OUD Instanz *oud\_adproxy* durch Aufrufen von 01\_create\_eus\_proxy\_instance.sh. Das Wrapper Skript erstellt mit *oud-proxy-setup* eine OUD Proxy Instanz mit AD Integration. Schauen Sie sich das Script vor der Ausführung kurz an.

```
less /u01/admin/oud_adproxy/create/01_create_eus_proxy_instance.sh
/u01/admin/oud_adproxy/create/01_create_eus_proxy_instance.sh
```

## 5. Anpassen der Konfiguration für die Instanz *oud\_adproxy* durch Aufrufen von 02\_config\_eus\_context.sh. Das Wrapper Skript führt die *dsconfig* Kommandos aus der Datei *02\_config\_eus\_context.conf* im Batch Mode aus. Mit den *dsconfig* Kommandos werden die Workflows für die AD Integration angepasst sowie Transformation Workflows für die AD spezifischen Attribute erstellt. Im GUI Mode wird dies vom GUI direkt gemacht. Schauen Sie sich das Script und die Konfig Datei vor der Ausführung kurz an.

```
less /u01/admin/oud_adproxy/create/02_config_eus_context.conf
less /u01/admin/oud_adproxy/create/02_config_eus_context.sh
/u01/admin/oud_adproxy/create/02_config_eus_context.sh
```

## 6. Mit dem folgenden Script wird der Oracle Context angepasst. Es werden sowohl die Standardwerte für die Benutzer sowie Gruppen Suche festgelegt. Dieser Schritt muss unabhängig ob CLI oder GUI Mode immer manuell ausgeführt werden.

```
less /u01/admin/oud_adproxy/create/03_config_eus_realm.ldif
less /u01/admin/oud_adproxy/create/03_config_eus_realm.sh
/u01/admin/oud_adproxy/create/03_config_eus_realm.sh
```

## 7. Anpassen der OUD Instanz. Neben den verschiedenen Logger werden unter anderem zwei Punkte angepasst, welche in der MOS Notes 2001851.1 ausführlicher beschrieben wird.

```
less /u01/admin/oud_adproxy/create/04_config_oud_ad_proxy.conf
less /u01/admin/oud_adproxy/create/04_config_oud_ad_proxy.sh
/u01/admin/oud_adproxy/create/04_config_oud_ad_proxy.sh
```

8. Oracle Enterprise User Security respektive das `eusm` Tool führt jeweils eine SASL Authentifizierung aus. Damit dies auch mit dem `dbca` sowie `eusm` klappt, muss das Passwort des verwendeten Benutzers mit einem *reversible* Passwort Hash abgelegt sein. D.h. in dem Fall *AES*. Aktuell verwenden wir den Benutzer `cn=Directory Manager`. Im Folgenden Skript wird die Passwort Policy erweitert und anschliessend das Passwort vom Benutzer `cn=Directory Manager` neu gesetzt.

```
less /u01/admin/oud_adproxy/create/05_update_directory_manager.sh
/u01/admin/oud_adproxy/create/05_update_directory_manager.sh
```

9. Abschliessend legen wir noch weitere Admin benutzer für die OUD administration an.

```
less /u01/admin/oud_adproxy/create/06_create_root_users.ldif
less /u01/admin/oud_adproxy/create/06_create_root_users.sh
less /u01/admin/oud_adproxy/create/07_create_eusadmin_users.sh
/u01/admin/oud_adproxy/create/07_create_eusadmin_users.sh
/u01/admin/oud_adproxy/create/06_create_root_users.sh
```

Mit den Scripts haben wir nun ein OUD AD Proxy instanz erstellt. Sie können Sich den Inhalt mit dem Directory Manager oder im folgenden mit dem OUDSM.

## Oracle Unified Directory Services Manager

Oracle Unified Directory Services Manager (OUDSM) wird dient als Weboberfläche für die OUD Instanzen. Mit dem OUDSM sind eine einfache Administration sowie der Zugriff auf den LDAP Baum möglich. OUDSM wird mit einer entsprechenden Python Procedure und dem `wlst.sh` Tool erstellt. Das passende Python Script `create_oudsm_domain.py` steht im Verzeichnis `$cdl/doag2018/lab/05_oud/` zur Verfügung.

1. Setzen eines Admin Passwortes in `create_oudsm_domain.py`

```
sed -i -e "s|ADMIN_PASSWORD|LAB01schulung|g" $cdl/doag2018/lab/05_oud/
create_oudsm_domain.py
```

2. Oracle Home auf die Oracle collocated Installation setzten und die OUDSM Domain erstellen.

```
export DOMAIN_NAME="oudsm_domain"
```

```
export DOMAIN_HOME="/u01/domains/oudsm_domain"
export PORT=7001
export PORT_SSL=7002
export ADMIN_USER="weblogic"
export ORACLE_HOME="/u00/app/oracle/product/fmw12.2.1.3.0"

${ORACLE_HOME}/oracle_common/common/bin/wlst.sh \
    -skipWLSModuleScanning $cdl/doag2018/lab/05_oud/create_oudsm_domain.py
```

3. OUDTAB Eintrag für OUDSM erstellen und Umgebung mit OUDBase setzen.

```
echo "oudsm_domain:7001:7002::OUDSM:N" >> ${ETC_BASE}/oudtab
. oudenv.sh oudsm_domain
```

4. Starten von OUDSM mit nohup. Nach einigen Minuten kann vom Active Directory Server auf die URL <http://oud.trivadislabs.com:7001/oudsm> vom OUDSM zugegriffen werden. Status im OUDSM Logfile respektive nohup.out File muss auf *RUNNING* stehen.

```
. oudenv.sh oudsm_domain
nohup /u01/domains/oudsm_domain/startWebLogic.sh &
```

## Zusatzaufgaben: Administration, Hochverfügbarkeit und Backup & Recovery

Stoppen und Starten einer OUD Instanz.

```
. oudenv.sh oud_ad
stop-ds

start-ds
```

Anzeigen des Instanz Status. Mit OUD Base Alias *u* respektive *oud\_up* oder mit dem OUD *status* Tool.

```
u
oud_up
. oudenv.sh oud_doag
status

status --bindDN "cn=Directory Manager" --bindPasswordFile $PWD_FILE
```

Anzeigen der definierten Backends zu einer OUD Instanz.

```
. oudenv.sh oud_doag  
list-backends
```

Sichern aller Backends mit `backup`. Dabei soll ein Fullbackup in das Verzeichnis `/u01/backup/oud_doag` gemacht werden. Zusätzlich wird das Backup noch komprimiert. Damit das Backup nicht interaktiv erstellt werden muss, wird zudem das Passwort in ein File abgespeichert.

```
echo "LAB01schulung" >/u01/admin/oud_doag/etc/oud_doag_pwd.txt  
chmod 600 /u01/admin/oud_doag/etc/oud_doag_pwd.txt  
. oudenv.sh oud_doag  
mkdir -p /u01/backup/oud_doag  
  
backup --bindPasswordFile $PWD_FILE \  
        --backUpAll --trustAll --compress \  
        --backupDirectory /u01/backup/oud_doag/
```

Alternativ lässt sich das gleiche Backup auch mit dem Skript `oud_backup.sh` aus OUD Base erstellen. Das Skript bietet zudem 2-3 zusätzliche Features wie Backup mehrerer Instanzen, Versand von e-Mails etc.

```
oud_backup.sh -h  
  
oud_backup.sh -v
```

Falls noch Zeit übrig ist, bieten sich folgende Zusatzaufgaben an:

- Anpassen weitere
- LDIF Export des ganzen Directories oder eines Teilbaumes
- Erstellen einer weiteren Instanz für den Aufbau einer Replikationsumgebung
  - Zweite Instanz anlegen analog `oud_doag` oder `oud_adproxy`. Entsprechend andere Ports und Instanz Name wählen
  - Konfiguration der Replikation mit `dsreplication` oder via OUDSM.
- Erstellen Sie mit `create-suffix` manuell einen EUS Suffix in der OUD Instanz. Was benötigt man noch, damit dieser Suffix auf für EUS verwendet werden kann.

## Übungen: Oracle Enterprise User Security

**Übungsziele:** Konfiguration von Enterprise User Security auf dem Datenbank Server. Erst von Mappings für verschiedenen Anwendungsfälle. Authentifizierung und Autorisierung mit Enterprise User Security sowie erfolgreichem Login mit Passwort sowie Kerberos Authentifizierung.

## SQLNet Konfiguration Enterprise User Security

Anpassen der SQLNet Konfiguration für die DB TDB122A respektive für alle Datenbanken auf diesem Server.

```
cp $cdl/doag2018/lab/06_eus/ldap.ora $cdn/admin/ldap.ora
vi $cdn/admin/ldap.ora

DIRECTORY_SERVERS = (oud.trivadislabs.com:1389:1636)
DEFAULT_ADMIN_CONTEXT = "dc=trivadislabs,dc=com"
DIRECTORY_SERVER_TYPE = OID
```

## Enterprise User Security Datenbank Konfiguration

Registrierung der DB TDB122A mit dem DBCA

```
TDB122A

dbca -configureDatabase -sourceDB $ORACLE_SID -registerWithDirService true \
    -dirServiceUserName "cn=Directory Manager" -dirServicePassword \
        LAB01schulung \
    -walletPassword LAB01schulung -silent
```

## Oracle Enterprise User Security Mappings

- Erstellen Sie ein shared global Schema EUS\_USERS

```
CREATE USER eus_user IDENTIFIED GLOBALLY;
GRANT connect TO eus_user;
GRANT SELECT ON v_$session TO eus_user;
```

- Erstellen Sie ein Mapping für die User in der Gruppe *Trivadis LAB Users*.

```
eusm createMapping database_name="$ORACLE_SID" \
    realm_dn="dc=trivadislabs,dc=com" map_type=SUBTREE \
    map_dn="ou=People,dc=trivadislabs,dc=com" schema=EUS_USERS \
    ldap_host="oud.trivadislabs.com" ldap_port=1389 ldap_user_dn="cn= \
        Directory Manager" \
    ldap_user_password="LAB01schulung"
```

- Testen Sie die Verbindung als Benutzer Blofeld.



```
CONNECT blofeld/LAB01schulung@TDB122A
```

```
show user
@sousrinf
SELECT * FROM session_roles;
```

- Erstellen Sie ein global private Schema

```
CONNECT / AS SYSDBA
CREATE USER king IDENTIFIED GLOBALLY;
GRANT connect TO king;
GRANT SELECT ON v_$session TO king;
```

- Erstellen Sie ein Mapping für den Benutzer King.

```
eusm createMapping database_name="$ORACLE_SID" \
    realm_dn="dc=trivadislabs,dc=com" map_type=ENTRY \
    map_dn="cn=Ben King,ou=Senior Management,ou=People,dc=trivadislabs,dc=com" \
    schema=KING \
    ldap_host="oud.trivadislabs.com" ldap_port=1389 \
    ldap_user_dn="cn=Directory Manager" \
    ldap_user_password="LAB01schulung"
```

- Testen Sie die Verbindung als Benutzer King.

```
CONNECT king/LAB01schulung@TDB122A
```

```
show user
@sousrinf
SELECT * FROM session_roles;
```

- Testen Sie die Verbindung als Benutzer King mit Kerberos

```
okinit king

sqlplus /@TDB122A

@sousrinf
SELECT * FROM session_roles;
```

- Prüfen Sie mit dem OUDSM oder Apache Directory Studio den Inhalt des OUD Proxy.

- Prüfen Sie die Mappings mit EUSM

```
eusm listMappings database_name="$ORACLE_SID" \  
  realm_dn="dc=trivadislabs,dc=com" \  
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \  
  ldap_user_dn="cn=Directory Manager" \  
  ldap_user_password="LAB01schulung"
```

## Oracle Enterprise User Security Rollen

- Erstellen Sie eine globale Datenbank Rolle hr\_clerk und hr\_mgr.

```
CONNECT / AS SYSDBA  
CREATE ROLE hr_clerk IDENTIFIED GLOBALLY;  
GRANT SELECT ON tvd_hr.employees TO hr_clerk;  
GRANT SELECT ON tvd_hr.jobs TO hr_clerk;  
GRANT SELECT ON tvd_hr.job_history TO hr_clerk;  
GRANT SELECT ON tvd_hr.regions TO hr_clerk;  
GRANT SELECT ON tvd_hr.countries TO hr_clerk;  
GRANT SELECT ON tvd_hr.locations TO hr_clerk;  
GRANT SELECT ON tvd_hr.departments TO hr_clerk;  
  
CREATE ROLE hr_mgr IDENTIFIED GLOBALLY;  
GRANT INSERT,UPDATE,DELETE ON tvd_hr.employees TO hr_mgr;  
GRANT INSERT,UPDATE,DELETE ON tvd_hr.jobs TO hr_mgr;  
GRANT INSERT,UPDATE,DELETE ON tvd_hr.job_history TO hr_mgr;  
GRANT INSERT,UPDATE,DELETE ON tvd_hr.locations TO hr_mgr;  
GRANT INSERT,UPDATE,DELETE ON tvd_hr.departments TO hr_mgr;
```

- Erstellen Sie eine Enterprise Rolle *HR Clerk* und *HR Management*.

```
eusm createRole enterprise_role="HR Clerk" \  
  domain_name="OracleDefaultDomain" \  
  realm_dn="dc=trivadislabs,dc=com" \  
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \  
  ldap_user_dn="cn=Directory Manager" \  
  ldap_user_password="LAB01schulung"  
  
eusm createRole enterprise_role="HR Management" \  
  domain_name="OracleDefaultDomain" \  
  realm_dn="dc=trivadislabs,dc=com" \  
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \  
  ldap_user_dn="cn=Directory Manager" \  
  ldap_user_password="LAB01schulung"
```

```
ldap_user_password="LAB01schulung"
```

- Zuweisen der Enterprise Rollen zu der globalen Rolle. Weisen Sie der Enterprise Rolle *HR Management* sowohl *hr\_clerk* also auch *hr\_mgr* zu.

```
eusm addGlobalRole enterprise_role="HR Clerk" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" database_name="$ORACLE_SID" \
  global_role="hr_clerk" dbuser="system" dbuser_password="LAB01schulung" \
  \
  dbconnect_string="db.trivadislabs.com:1521:$ORACLE_SID" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

eusm addGlobalRole enterprise_role="HR Management" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" database_name="$ORACLE_SID" \
  global_role="hr_clerk" dbuser="system" dbuser_password="LAB01schulung" \
  \
  dbconnect_string="db.trivadislabs.com:1521:$ORACLE_SID" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

eusm addGlobalRole enterprise_role="HR Management" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" database_name="$ORACLE_SID" \
  global_role="hr_mgr" dbuser="system" dbuser_password="LAB01schulung" \
  dbconnect_string="db.trivadislabs.com:1521:$ORACLE_SID" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"
```

- Weisen Sie die Rolle Benutzer zu. Erstellen Sie zuerst im AD eine neue Gruppe für HR. Anschliessend nehmen Sie die HR Gruppe für *HR Clerk* und *Honey Rider* für *HR Management*.

```
eusm grantRole enterprise_role="HR Clerk" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  group_dn="CN=Trivadis LAB HR,OU=Groups,DC=trivadislabs,DC=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
```

```

ldap_user_password="LAB01schulung"

eusm grantRole enterprise_role="HR Management" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  user_dn="CN=Honey Rider,OU=Human Resources,OU=People,DC=trivadislabs,DC=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

```

- Loggen Sie sich als *Honey Rider* und *Vesper Lynd* ein. Was haben Sie für Rollen?

```

CONNECT rider/LAB01schulung@TDB122A
SELECT sys_context('userenv','ENTERPRISE_IDENTITY') FROM DUAL;
SELECT * FROM session_roles;

CONNECT lynd/LAB01schulung@TDB122A
show user
SELECT sys_context('userenv','ENTERPRISE_IDENTITY') FROM DUAL;
SELECT * FROM session_roles;

```

- Was Stellen Sie fest? Stimmen die Rollen von *Vesper Lynd*?
- Weisen Sie im AD *Vesper Lynd* die Richtige Rolle zu und loggen sich sich erneut ein.

```

CONNECT lynd/LAB01schulung@TDB122A
show user
SELECT sys_context('userenv','ENTERPRISE_IDENTITY') FROM DUAL;
SELECT * FROM session_roles;

```

- Anzeigen der Enterprise Rollen mit EUSM

```

eusm listEnterpriseRoles domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

eusm listEnterpriseRoleInfo enterprise_role="HR Management" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

```

```

ldap_user_password="LAB01schulung"

eusm listEnterpriseRoleInfo enterprise_role="HR Clerk" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

```

## Oracle Enterprise User Security Proxy Authentifizierung

Die Entwickler aus der Entwicklungsabteilung sollen das Recht Erhalten sich als Proxy auf das TVD\_HR Schema anzumelden.

- Erstellen der Proxy Berechtigung in der Datenbank-

```

CONNECT / AS SYSDBA
ALTER USER tvd_hr GRANT CONNECT THROUGH ENTERPRISE USERS;

```

- Erstellen des Enterprise Proxy Recht

```

eusm createProxyPerm proxy_permission="TVD HR Devel" \
  domain_name="OracleDefaultDomain" \
  realm_dn="dc=trivadislabs,dc=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

```

- Weisen Sie das Proxy Recht der AD Gruppe *Trivadis LAB Developers* zu.

```

eusm grantProxyPerm proxy_permission="TVD HR Devel" \
  domain_name="OracleDefaultDomain" \
  group_dn="CN=Trivadis LAB Developers,OU=Groups,DC=trivadislabs,DC=com" \
  realm_dn="dc=trivadislabs,dc=com" \
  ldap_host="oud.trivadislabs.com" ldap_port=1389 \
  ldap_user_dn="cn=Directory Manager" \
  ldap_user_password="LAB01schulung"

```

- Prüfen Sie die Verbindung normal und als proxy mit einem Mitarbeiter aus dem Development Team. *Douglas Adams, Ernst Blofeld, Henry Ford, James Scott, Paul Smith.*

```
CONNECT ford/LAB01schulung@TDB122A
show user
@sousrinf
SELECT * FROM session_roles;

CONNECT blofeld[tvd_hr]/LAB01schulung@TDB122A
show user
@sousrinf
SELECT * FROM session_roles;
```

## Demo- und Lab Umgebung

issue und manuelle konfig \* dns \* resolve dns search \* putty / ssh keys \* oracle vagrant gruppe

### Architektur

Für die praktischen Arbeiten im Rahmen des DOAG 2018 Schulungstages, steht pro zweiter Team eine einfach Testumgebung zur Verfügung. Die Umgebung läuft für die Dauer der Schulung in der [Oracle Ravello Cloud](#) und besteht, wie in der folgenden Abbildung ersichtlich aus folgenden Servern respektive VMs:

- **db.trivadislabs.com** Oracle Datenbank Server mit Oracle 12c R2 sowie 18c
- **oud.trivadislabs.com** Oracle Directory Server mit Oracle Unified Directory 12c
- **ad.trivadislabs.com** MS Windows Server 2012 R2 mit Active Directory

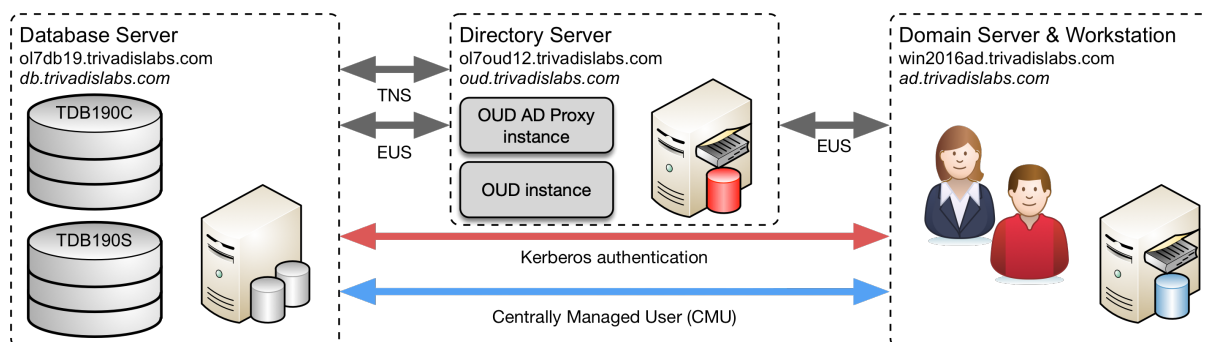


Abb. 2: Architektur Schulungsumgebung

Die Umgebung ist soweit vorbereitet, dass direkt mit den Übungen gestartet werden kann.

Die zentrale Benutzerverwaltung mit *Oracle Centrally Managed Users* oder *Oracle Enterprise User Security* sind komplexe Themen, welche nicht abschliessend am Schulungstag diskutiert werden können. Aus diesem Grund gibt es für das Selbststudium die Möglichkeit, eine Testumgebung analog

dem Schultag aufzubauen. Diese Umgebung wird Skript gestützt mit [Vagrant](#) auf [Virtualbox](#) aufgebaut. Man benötigt lediglich die entsprechenden Software Images für die Oracle Datenbank 12c R2 + 18c, Oracle Unified Directory sowie die Umgebungsskripte. Anschliessend lässt sich die Umgebung nahezu voll automatisch aufbauen. Eine entsprechende Anleitung für den Aufbau der Trivadis LAB Umgebung sowie die dazugehörigen *Vagrant Files*, *Skripte* etc. findet man im GitHub Repository [oehrli/trivadislabs.com](https://github.com/oehrli/trivadislabs.com).

## Oracle Datenbank Server

### Generelle Server Konfiguration

Der Oracle Datenbank Server ist wie folgt konfiguriert:

- **Host Name** : db.trivadislabs.com
- **Interne IP Adresse** : 10.0.0.3
- **Externe IP Adresse** : gemäss Liste
- **Betriebssystem** : Oracle Enterprise Linux Server Release 7.5
- **Oracle Datenbank Software** :
  - Oracle 12c Release 2 Enterprise Edition (12.2.0.1) mit Release Update vom Oktober 2018
  - Oracle 18c Enterprise Edition (18.4.0.0) mit Release Update vom Oktober 2018
- **Oracle Datenbanken** :
  - **TDB122A** Oracle 12cR2 Enterprise Edition Single Instance für die Übungen mit EUS
  - **TDB184A** Oracle 18c Enterprise Edition Single Instance für die Übungen mit CMU
- **Betriebssystem Benutzer** :
  - oracle / PASSWORT
  - root / PASSWORT
- **Datenbank Benutzer** :
  - sys / manager
  - system / manager
  - scott / tiger
  - tvd\_hr / tvd\_hr

### Trivadis BasEnv

Das Trivadis Base Environment (TVD-Basenv™) ermöglicht einfaches Navigieren in der Directory Struktur und zwischen den verschiedenen Datenbanken. In der folgenden Tabelle sind die Aliases für den OS Benutzer *oracle* aufgelistet, welche am häufigsten verwendet werden.

Alias Name	Beschreibung
cda	zum Admin Verzeichnis der aktuell gesetzten Datenbank
cdh	zum Oracle Home
cdob	zum Oracle Base
cdt	zum TNS_ADMIN
sqh	startet SQLPlus mit „sqlplus / as sysdba“ inklusive Befehlshistory
sta	Statusanzeige für die aktuell gesetzte Datenbank
taa	öffnet das Alertlog der aktuell gesetzten Datenbank mit <code>tail -f</code>
TDB122A	setzt die Umgebung im Terminal für die Datenbank <i>TDB122A</i>
TDB184A	setzt die Umgebung im Terminal für die Datenbank <i>TDB184A</i>
u	Statusanzeige für alle Oracle Datenbanken und Listener (z.B. open, mount)
via	öffnet das Alertlog der aktuell gesetzten Datenbank in vi

Die Installation ist nach dem OFA (Optimal Flexible Architecture) Standard vorgenommen worden – Beispiel für die Installation auf der Datenbank-VM für die Datenbank - *TDB122A*:

Mount Point / Directory	Beschreibung
<code>/u00/app/oracle/admin/TDB122A/adump</code>	Oracle Audit Files
<code>/u00/app/oracle/admin/TDB122A/backup</code>	Oracle Backup
<code>/u00/app/oracle/admin/TDB122A/dpdump</code>	Data Pump Dateien
<code>/u00/app/oracle/admin/TDB122A/etc</code>	Oracle Backup Konfig Dateien
<code>/u00/app/oracle/admin/TDB122A/log</code>	Log Dateien (z.B. Backup, Export, etc.)
<code>/u00/app/oracle/admin/TDB122A/pfile</code>	Parameter- und Password-Datei
<code>/u00/app/oracle/admin/TDB122A/wallet</code>	Oracle Wallet
<code>/u00/app/oracle/etc</code>	oratab und diverse Konfigurationsdateien
<code>/u00/app/oracle/local/dba</code>	Environment Tools (TVD-Basenv)
<code>/u00/app/oracle/network/admin</code>	Oracle Net Konfigurationsdateien
<code>/u00/app/oracle/product/12.2.0.1</code>	Oracle 12.2.0.1 Home
<code>/u00/app/oracle/product/18.4.0.0</code>	Oracle 18.4.0.0 Home



Mount Point / Directory	Beschreibung
/u01/oradata/TDB122A	Datenbank Dateien, Redo Log Files, CTL
/u02/fast_recovery_area/TDB122A	Fast Recovery Area
/u02/oradata/TDB122A	Redo Log Files, CTL

## Übungschema TVD\_HR

In den Datenbanken ist neben dem Scott Demo Schema zusätzlich das Beispiel Schema *TVD\_HR*. Das Schema *TVD\_HR* basiert auf dem bekannten Oracle *HR* Beispiel Schema. Der wesentliche Unterschied zum regulären *HR* Schema ist, dass die Abteilungen sowie Mitarbeiter den Mitarbeitern im Active Directory entspricht.

Erklärung zu den Tabellen basierend auf den Kommentaren vom *HR* Schema:

- **REGIONS** Tabelle, welche Regionsnummern und -namen enthält. Verweise auf die Tabelle *LOCATION*.
- **LOCATIONS** Tabelle, die die spezifische Adresse eines bestimmten Büros, Lagers und/oder Produktionsstandortes eines Unternehmens enthält. Speichert keine Adressen von Kundenstandorten.
- **DEPARTMENTS** Tabelle, die Details zu den Abteilungen zeigt, in denen die Mitarbeiter arbeiten. Verweise auf Standorte, Mitarbeiter und Job History Tabellen.
- **JOB\_HISTORY** Tabelle, in der die Beschäftigungshistorie der Mitarbeiter gespeichert ist. Wenn ein Mitarbeiter innerhalb der Stelle die Abteilung wechselt oder die Stelle innerhalb der Abteilung wechselt, werden neue Zeilen in diese Tabelle mit alten Stelleninformationen des Mitarbeiters eingefügt. Verweise auf Tabellen mit Jobs, Mitarbeitern und Abteilungen.
- **COUNTRIES** Tabelle. Verweise mit der Tabelle der Standorte.
- **JOBS** Tabelle mit Jobbezeichnungen und Gehaltsgruppen. Verweise auf Mitarbeiter und Job History Tabelle.
- **EMPLOYEES** Tabelle. Verweise mit Abteilungen, Jobs, Job History Tabellen. Enthält eine Selbstreferenz.

Zukünftige Versionen von TVD\_HR werden zusätzlich entsprechend VPD Policies enthalten.

## Oracle Unified Directory Server

### Generelle Server Konfiguration

Der Directory Server ist wie folgt konfiguriert:

- **Host Name** : oud.trivadislabs.com
- **Interne IP Adresse** : 10.0.0.5
- **Externe IP Adresse** : gemäss Liste
- **Betriebssystem** : Oracle Enterprise Linux Server Release 7.5
- **Java** : Oracle JAVA Server JRE 1.8 u192
- **Oracle Fusion Middleware Software** :
  - Oracle Unified Directory (12.2.1.3) mit dem Bundle Patch vom Oktober 2018
  - Oracle Fusion Middleware Infrastructure Directory (12.2.1.3) mit dem Bundle Patch vom Oktober 2018
- **Oracle Home oud12.2.1.3** : Oracle Unified Directory *standalone* Installation.
- **Oracle Home fmw12.2.1.3** : Oracle Unified Directory *collocated* Installation mit Oracle Fusion Middleware Infrastructure.
- **Betriebssystem Benutzer** :
  - oracle / PASSWORT
  - root / PASSWORT

## Trivadis OUD Base

Analog zu der Datenbank Umgebung, gibt es auch für Oracle Unified Directory entsprechende Umgebungsscripte. Diese Umgebungsscripte, kurz auch OUD Base genannt, werden unter anderem in [OUD Docker images](#) verwendet. Aus diesem Grund ist OUD Base etwas "leichter" aufgebaut als TVD-Basenv und basiert zu 100% auf Bash. OUD Base ist via GitHub Projekt [oehrlis/oudbase](#) als Open Source verfügbar.

In der folgenden Tabelle sind die Aliases für den OS Benutzer *oracle* aufgelistet, welche am häufigsten verwendet werden.

Alias Name	Beschreibung
cda	zum Admin Verzeichnis der aktuell OUD Instanz
cdh	zum Oracle Home
cdih	zum OUD Instanz Home Verzeichnis
cdil	zum OUD Instanz Log Verzeichnis
cdob	zum Oracle Base
dsc	aufruf von dsconfig inklusive Host Name, <code>\$PORT_ADMIN</code> und <code>\$PWD_FILE</code>
oud_ad	setzt die Umgebung im Terminal für die OUD Instanz <i>oud_ad</i>
taa	öffnet das Access Log der aktuell gesetzten OUD Instanz mit <code>tail -f</code>

Alias Name	Beschreibung
u	Statusanzeige für alle OUD Instanz inkl entsprechender Ports
version	Anzeigen der Version von OUD base inklusive geänderten Dateien in <code>\$OUD_LOCAL</code>
vio	öffnet die oudtab Datei. <code>\${ETC_BASE}/oudtab</code>

Die Installation ist an den OFA (Optimal Flexible Architecture) Standard angelegt. Die Software, Konfiguration sowie Instanzen werden explizit von einander getrennt. Beispiel für die Installation auf der OUD-VM für die OUD Instanz - *oud\_ad*:

Mount Point / Directory	Beschreibung
<code>/u00/app/oracle/local/oudbase</code>	Environment Tools (OUD Base)
<code>/u00/app/oracle/product/fmw12.2.1.3.0</code>	Oracle Unified Directory 12.2.1.3 Collocated Home
<code>/u00/app/oracle/product/jdk1.8.0_192</code>	Oracle Java 1.8 update 192
<code>/u00/app/oracle/product/oud12.2.1.3.0</code>	Oracle Unified Directory 12.2.1.3 Standalone Home
<code>/u01/admin/oud_ad</code>	Instance Admin Verzeichnis
<code>/u01/backup</code>	Standard Backup Verzeichnis
<code>/u01/etc</code>	oudtab und diverse Konfigurationsdateien
<code>/u01/instances/oud_ad/OUUD/config</code>	Instanz Konfigurations Verzeichnis
<code>/u01/instances/oud_ad/OUUD/logs</code>	Instanz Log Verzeichnis
<code>/u01/instances/oud_ad</code>	Instanz Home Verzeichnis

## MS Active Directory Server

### Generelle Server Konfiguration

Der Active Directory Server basiert auf einer Windows Server 2012 R2 Umgebung (Windows Server 2016 für on-premises Setup) und ist wie folgt konfiguriert:

- **Host Name** : ad.trivadislabs.com
- **Interne IP Adresse** : 10.0.0.4

- **Externe IP Adresse** : gemäss Liste
- **Betriebssystem** : MS Windows Server 2012 R2
- **Installiere Server Roles** :
  - Active Directory Server
  - DNS Server mit Active Directory Integration
  - Certification Authority
- **Zusatz Software** : nur auf der Cloud VM
  - Putty für SSH Verbindungen mit dem OUD und DB Server
  - MobaXTerm für SSH Verbindungen mit dem OUD und DB Server
  - WinSCP für den File Transfer DB Server <=> AD Server
  - SQL Developer
  - Oracle 12c R2 und 18c Clients
  - MS Visual Studio Code als universellen Texteditor
  - Predefined SSH Keys für den OUD und DB Server
- **Betriebssystem Benutzer** :
  - Administrator / PASSWORT
  - root / PASSWORT
  - Trivadis LAB User / LAB01schulung

## AD Domain TRIVADISLAB

Damit eine mehr oder weniger praxis nahe Anbindung an das Active Directory möglich ist, wurde für die fiktive Firma *Trivadis LAB* eine einfache AD Struktur aufgebaut. Die folgende Abbildung zeigt das Organigramm inklusive Abteilungen und Mitarbeiter für *Trivadis LAB*. Sämtlich aufgeführte Benutzer können als Testbenutzer verwendet werden. Wobei der Loginname jeweils dem klein geschriebenen Nachname entspricht. Passwort ist für alle Benutzer *LAB01schulung*.

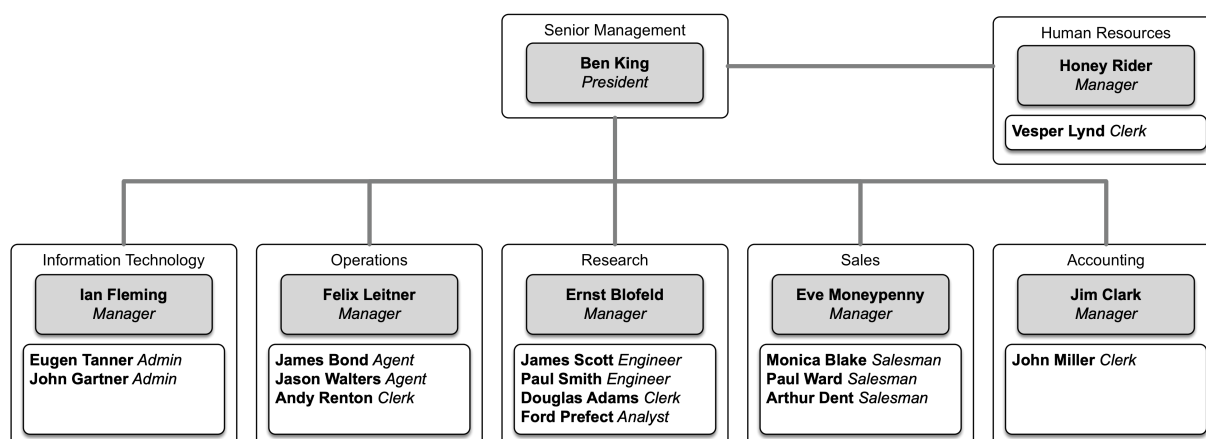


Abb. 3: Organigramm Trivadis LAB Company

Das fiktive Unternehmen hat folgende Abteilungen:

ID	Abteilung	Distinguished Name (DN)	Beschreibung
10	Senior Management	<code>ou=Senior Management,ou=People,dc=trivadislabs,dc=com</code>	Geschäftsleitung
20	Accounting	<code>ou=Accounting,ou=People,dc=trivadislabs,dc=com</code>	Finanzen
30	Research	<code>ou=Research,ou=People,dc=trivadislabs,dc=com</code>	Forschung
40	Sales	<code>ou=Sales,ou=People,dc=trivadislabs,dc=com</code>	Verkauf + Vertrieb
50	Operations	<code>ou=Operations,ou=People,dc=trivadislabs,dc=com</code>	Betriebsabteilung
60	Information Technology	<code>ou=Information Technology,ou=People,dc=trivadislabs,dc=com</code>	IT Abteilung
70	Human Resources	<code>ou=Human Resources,ou=People,dc=trivadislabs,dc=com</code>	Personalabteilung

Zusätzlich wurden folgende Gruppen definiert:

Gruppe	Distinguished Name (DN)	Beschreibung
Trivadis LAB APP Admins	<code>ou=Trivadis LAB APP Admins,ou=Groups,dc=trivadislabs,dc=com</code>	Applikations Administratoren
Trivadis LAB DB Admins	<code>ou=Trivadis LAB DB Admins,ou=Groups,dc=trivadislabs,dc=com</code>	DB Admins aus der IT Abteilung
Trivadis LAB Developers	<code>ou=Trivadis LAB Developers,ou=Groups,dc=trivadislabs,dc=com</code>	Entwickler aus der Forschungsabteilung
Trivadis LAB Management	<code>ou=Trivadis LAB Management,ou=Groups,dc=trivadislabs,dc=com</code>	Geschäftsleitung und Manager
Trivadis LAB System Admins	<code>ou=Trivadis LAB System Admins,ou=Groups,dc=trivadislabs,dc=com</code>	System Admins aus der IT Abteilung
Trivadis LAB Users	<code>ou=Trivadis LAB Users,ou=Groups,dc=trivadislabs,dc=com</code>	Alle Benutzer

## Links und Referenzen

### OOD EUS Workshop

Unterlagen und Skripte zum Workshop

- Übungsskripte zum DOAG Schulungstag [doag2018](#)
- Vagrant Setup zum Aufbau der Trivadis LAB Umgebung [oehrlis/trivadislabs.com](#)
- Setup Skripte für die Konfiguration der Umgebung (Cloud, Vagrant, Docker) [oehrlis/oradba\\_init](#)
- OUD Base Umgebungsskripte für Oracle Unified Directory [oehrlis/oudbase](#)

### Oracle Dokumentation

- Oracle Online Dokumentation 18c <https://docs.oracle.com/en/database/oracle/oracle-database/18/books.html>
- Oracle Enterprise User Security <https://docs.oracle.com/en/database/oracle/oracle-database/18/dbimi/index.html>
- Oracle Centrally Managed User [https://docs.oracle.com/en/database/oracle/oracle-database/18/dbseg/integrating\\_mads\\_with\\_oracle\\_database.html](https://docs.oracle.com/en/database/oracle/oracle-database/18/dbseg/integrating_mads_with_oracle_database.html)
- Oracle EUSM Utility <https://docs.oracle.com/en/database/oracle/oracle-database/18/dbimi/enterprise-user-security-manager-eusm-command-summary.html>

## Software und Tools

### Betriebssystem und Virtualisierung

- Oracle VM Virtualbox [virtualbox](#)
- HashiCorp Vagrant [vagrant](#)
- Oracle Enterprise Linux 7.5
  - Oracle Vagrant Boxes [vagrant image](#). Predefined Image von Oracle für die Nutzung mit Virtualbox und Vagrant. Das Vagrant Image wird bei einem [vagrant up](#) falls nicht vorhanden direkt herunter geladen.
  - Oracle Software Delivery Cloud [iso](#). Basis Setup iso File, falls individuell ein Oracle Linux Server installiert werden soll.
- Microsoft Windows Server 2016
  - Vagrant Box [StefanScherer/windows\\_2016](#). Vagrant Image aus der Vagrant Cloud. Erstellt von Stefan Scherer für die Nutzung mit Virtualbox und Vagrant. Das Vagrant Image wird bei einem [vagrant up](#) falls nicht vorhanden direkt herunter geladen.
  - Evaluation 2016 [iso](#). Basis Setup iso File, falls individuell ein Windows Server installiert werden soll.

- Trivadis BasEnv Test [basenv-18.05.final.b.zip](#)

### Oracle Datenbank Binaries

- Oracle Base Releases 12c Release 2 und 18c [Oracle Technology Network](#)
- Oktober Critical Patch Update Oracle Database 18c
  - DATABASE RELEASE UPDATE 18.4.0.0.0 [28655784](#)
  - OJVM RELEASE UPDATE: 18.4.0.0.181016 [28502229](#)
- Oktober Critical Patch Update Oracle Database 12c Release 2
  - DATABASE OCT 2018 RELEASE UPDATE 12.2.0.1.181016 [28662603](#)
  - OJVM RELEASE UPDATE 12.2.0.1.181016 [28440725](#)
- Oracle OPatch Utility 12.2.0.1.13 for DB 12.2.0.x and DB 18.x [6880880](#)

### Oracle Unified Directory Binaries

- Java Server 1.8 u192 [28414856](#)
- Oracle Fusion Middleware 12.2.1.3.0 Oracle Unified Directory [26270957](#)
- OUD BUNDLE PATCH 12.2.1.3.0(ID:180829.0419) [28569189](#)
- Oracle Fusion Middleware 12.2.1.3.0 Fusion Middleware Infrastructure [26269885](#)
- WLS PATCH SET UPDATE 12.2.1.3.181016 Oracle WLS 12.2.1.3.0 [28298734](#)
- OPatch Utility für WLS [28186730](#)
- OUD Base Umgebungsskripte für Oracle Unified Directory [oehrlis/oudbase](#)

### Tools Active Directory Server

- Oracle Clients
  - Oracle Clients [Oracle Technology Network](#)
  - Oracle Instant Clients [Oracle Technology Network](#)
- Apache Directory Studio LDAP Browser [Home](#)
- Putty SSH Utility [Putty Home](#)
- WinSCP SFTP client und FTP Client für Microsoft Windows [WinSCP Home](#)