TDM (Test Data Management) Approach to sync Customer Env. for Validation & Perform Environment Check - Identity Suite vApp Example

Alan Baugher, CA Sr. Principal Architect

Jan, 2017

ca technologies

Process

Create MS Windows Service ID & Network Share
- Used to copy business logic from solution to remote location

Enable SAMBA Network Share on Identity Suite vApp

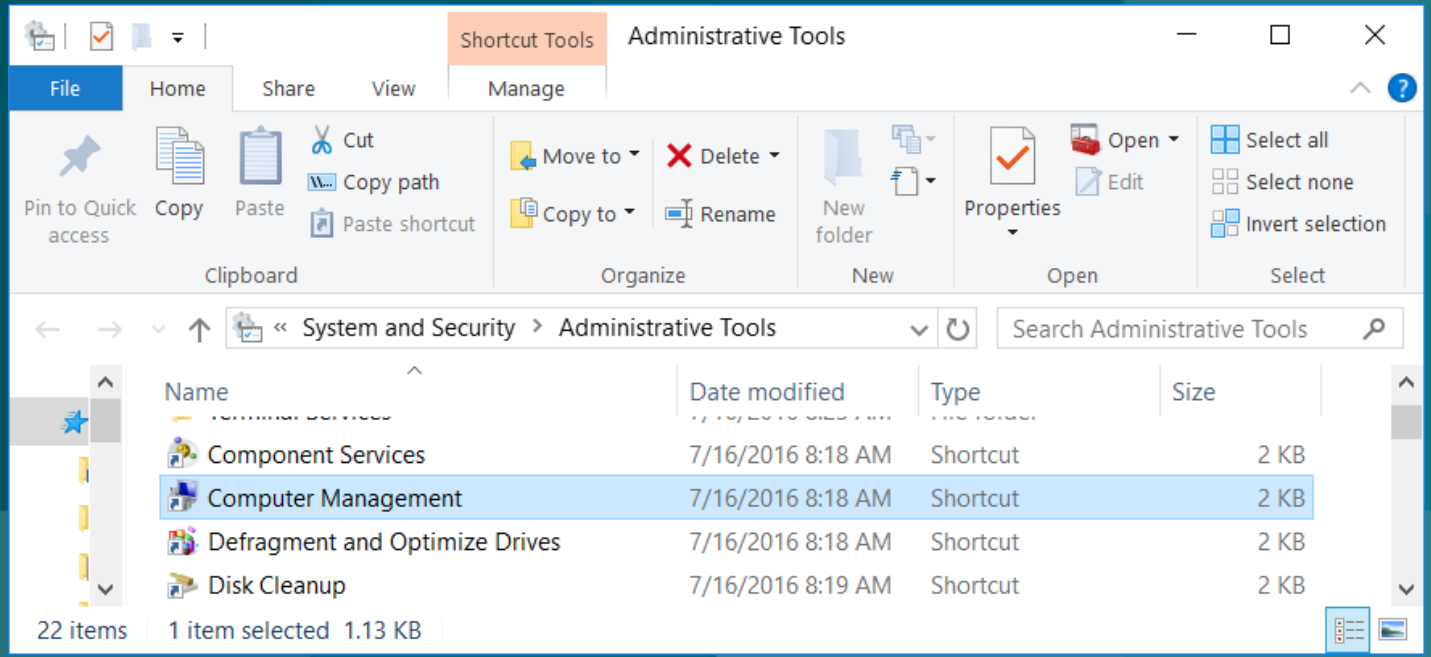
Export IM Business Content using new IM r14 export tool
- ImportExportUtility

Export IM Userstore/Provisioning Store
- Full Data Extract, TDM Extract (No Passwords/No Policies)

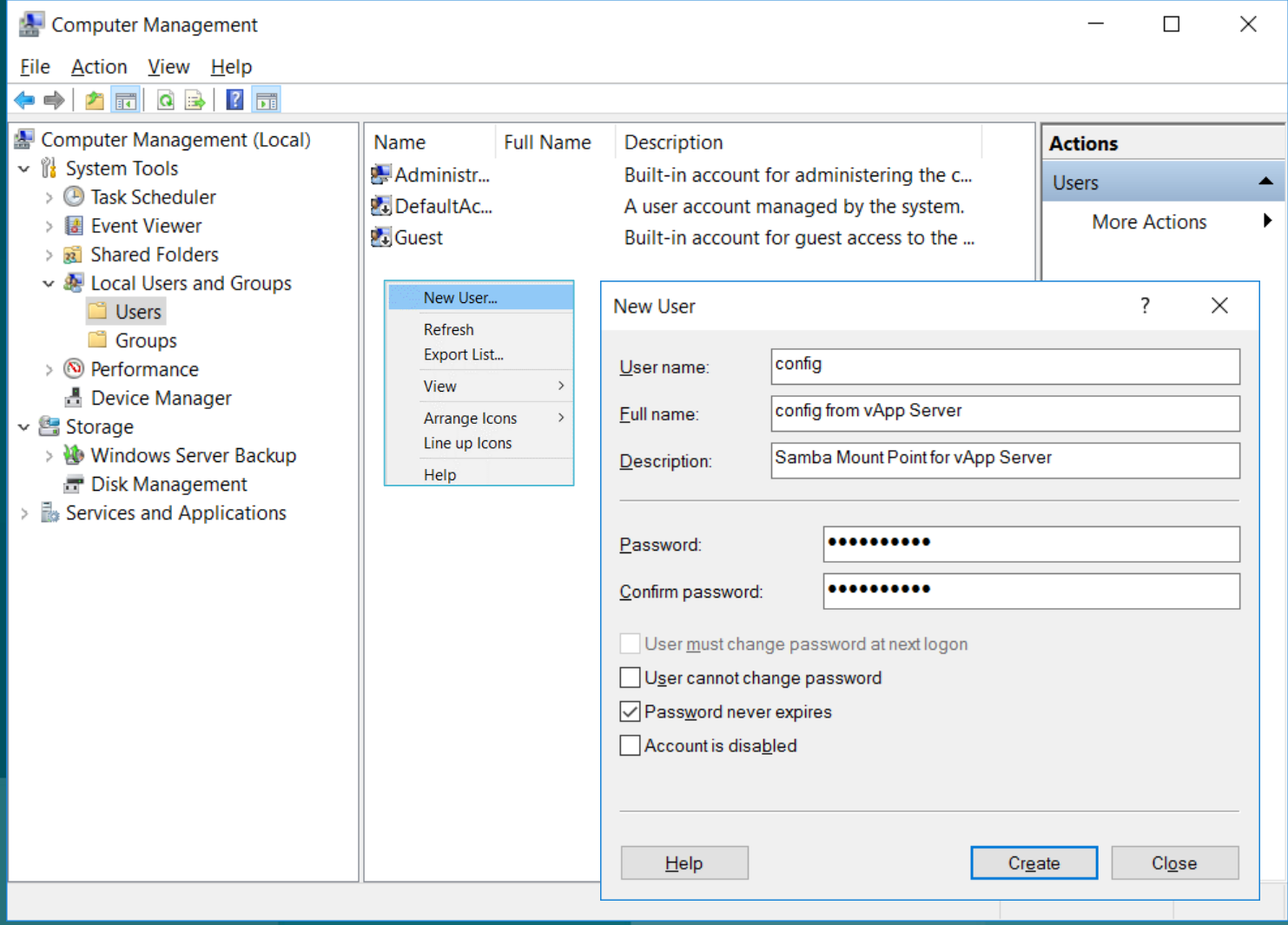
Copy Data to SAMBA Network Share

Create a new Service ID

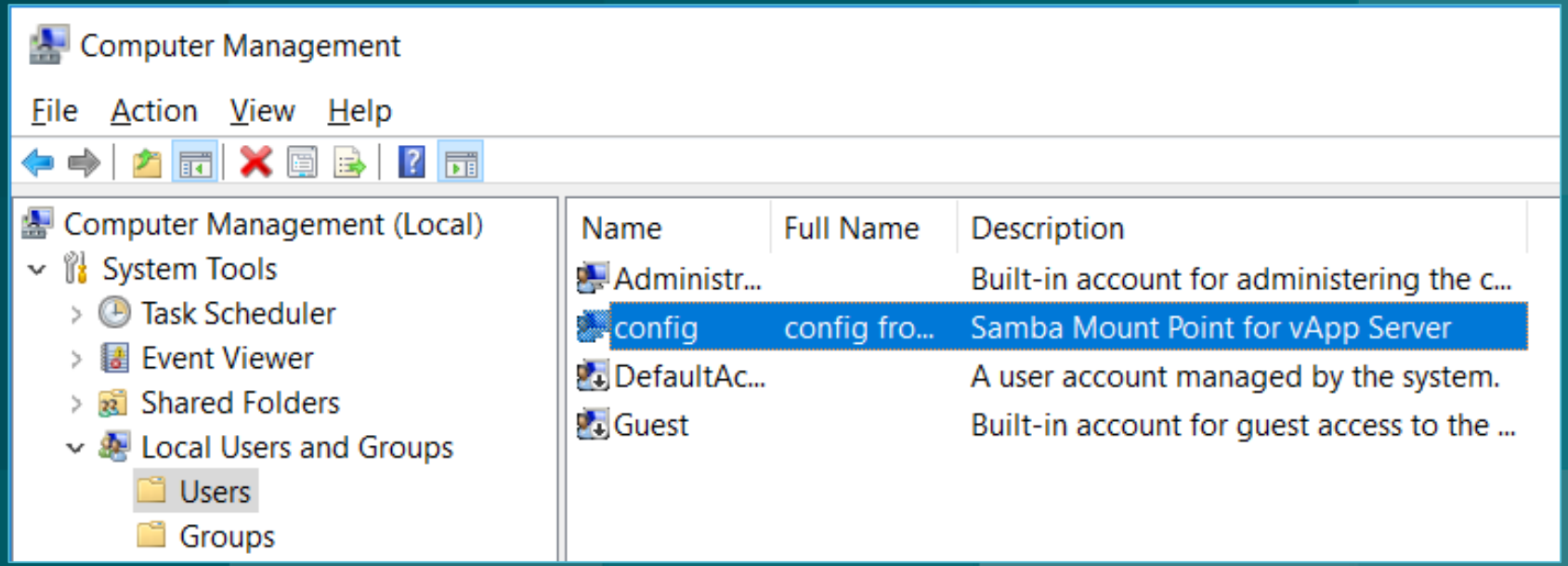
- Open the MS Windows Administrative Tools: Computer Management



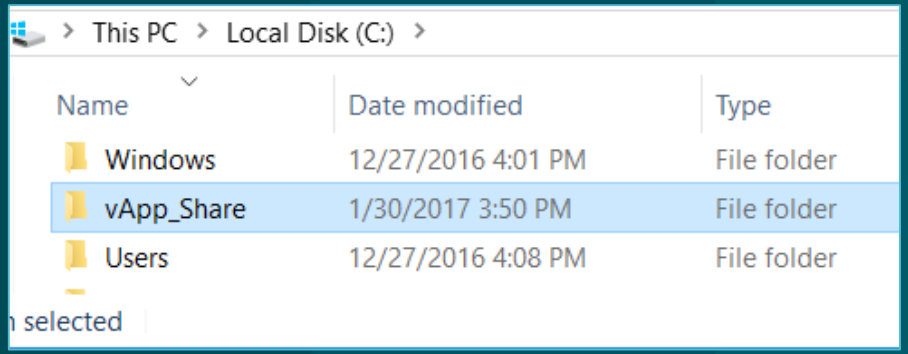
1. Select Local User & Groups
2. Select Users.
3. Right click in middle large panel and select New User…

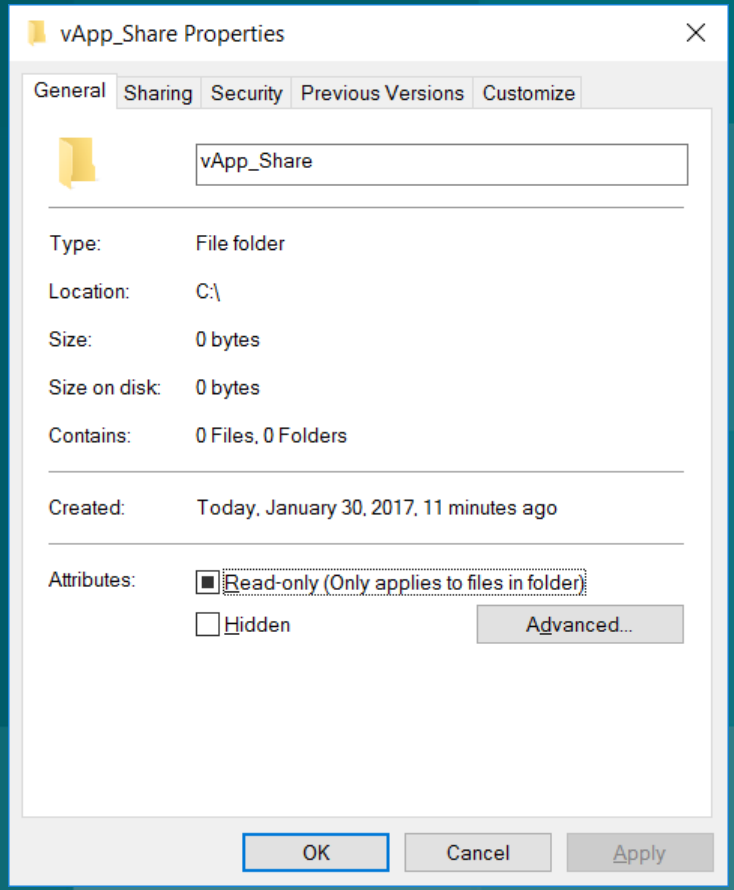


New Service ID created on local MS Windows Server

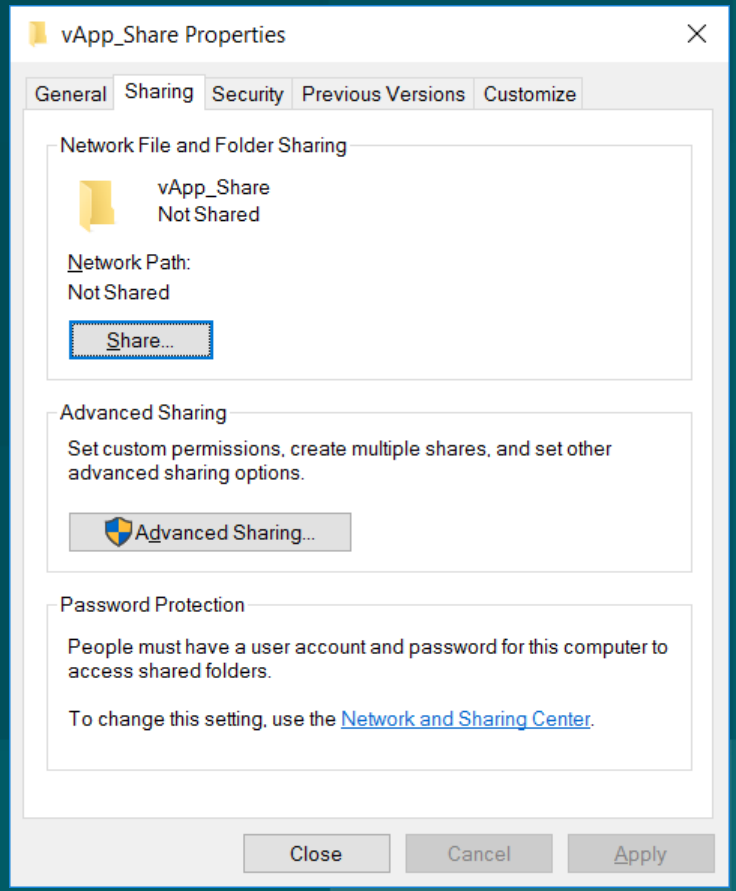


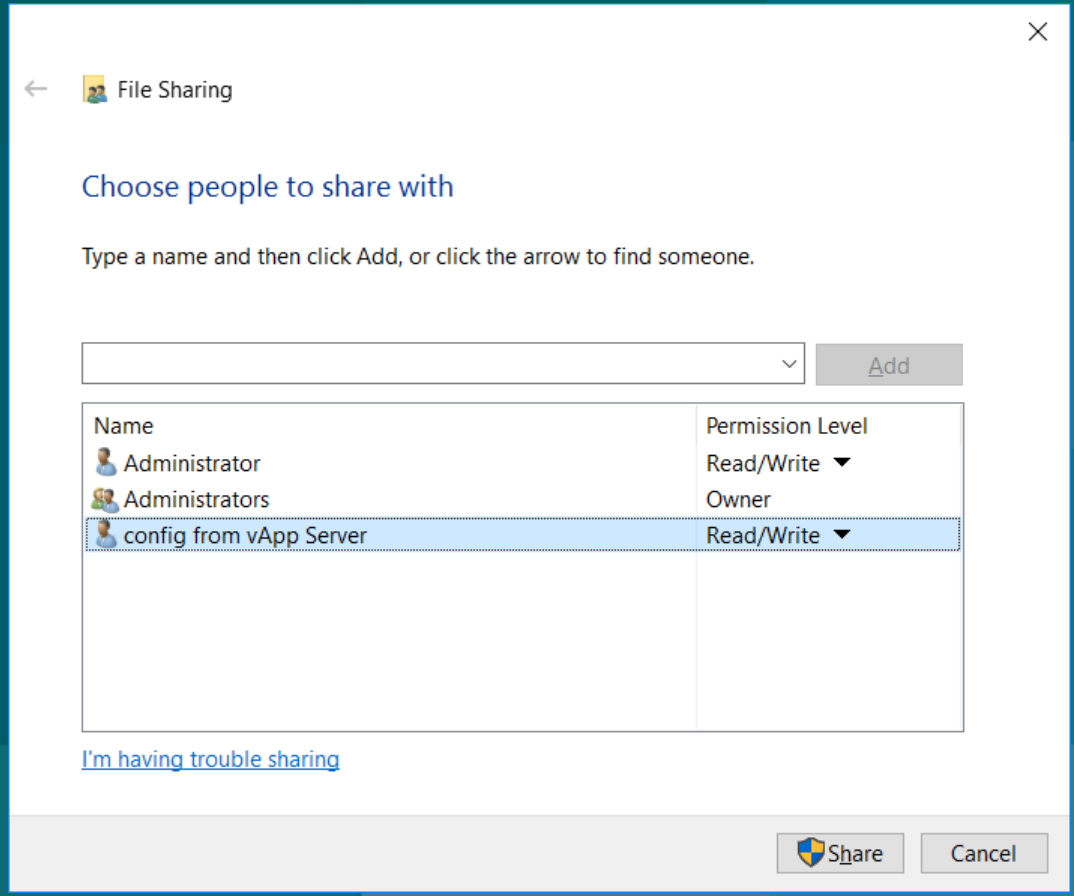
Create File Share on MS Windows Server

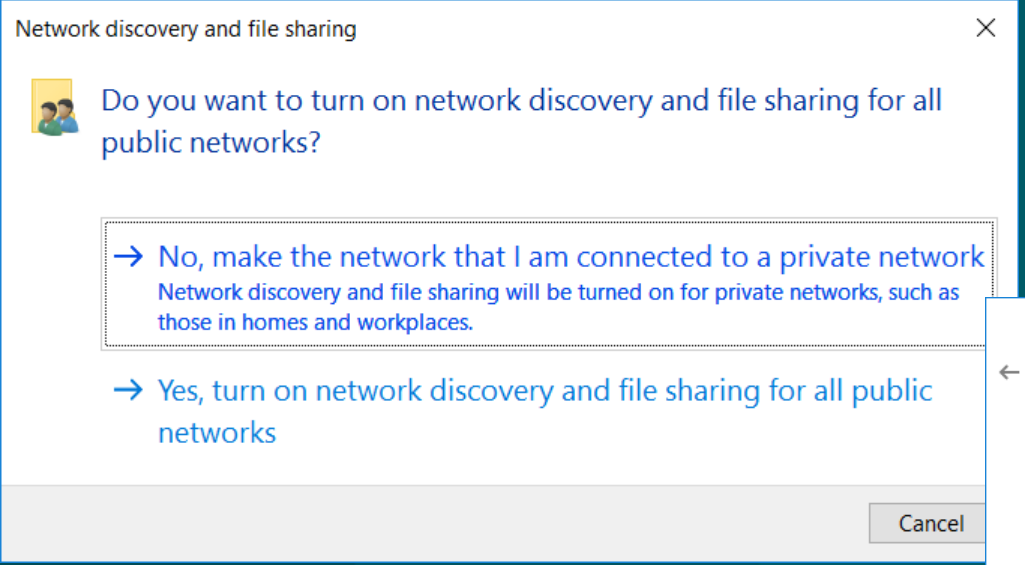


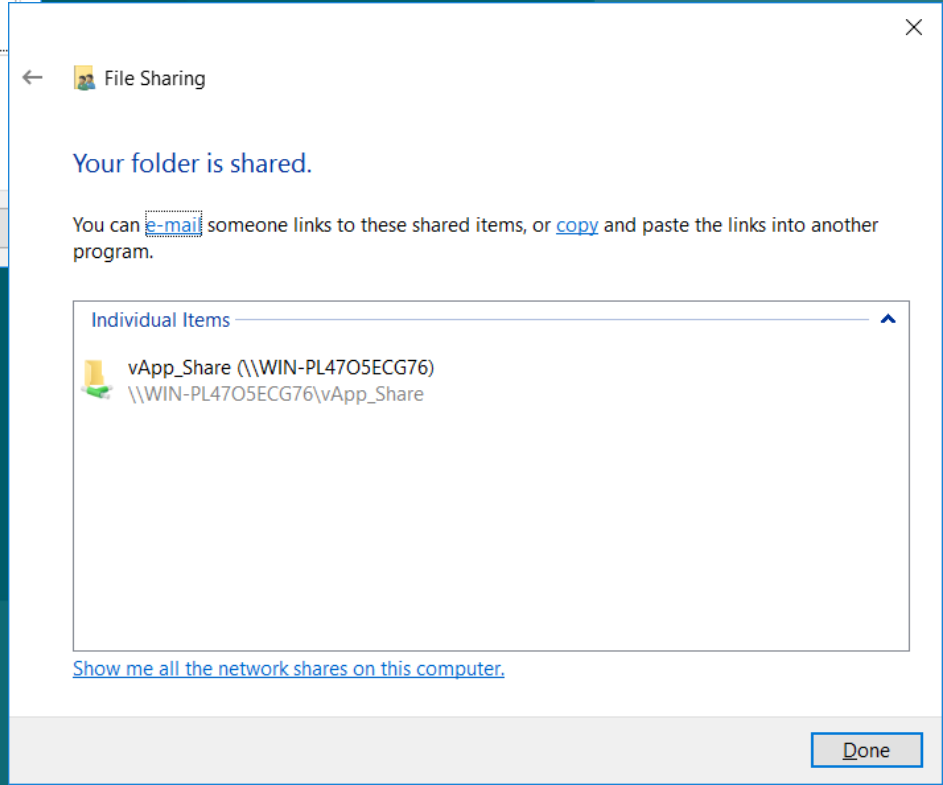


Share Folder with ‘config’ service ID

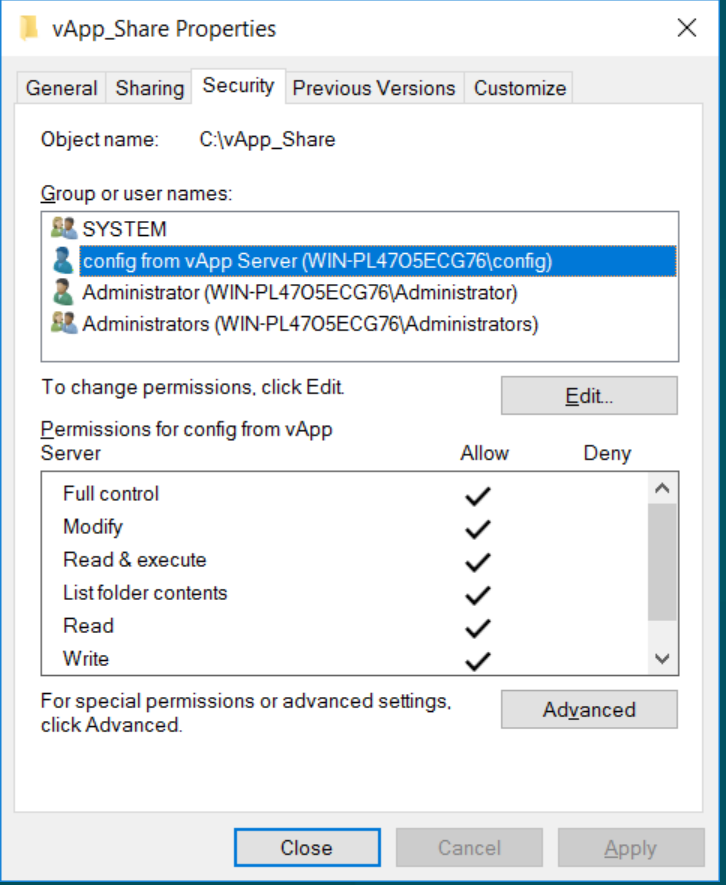




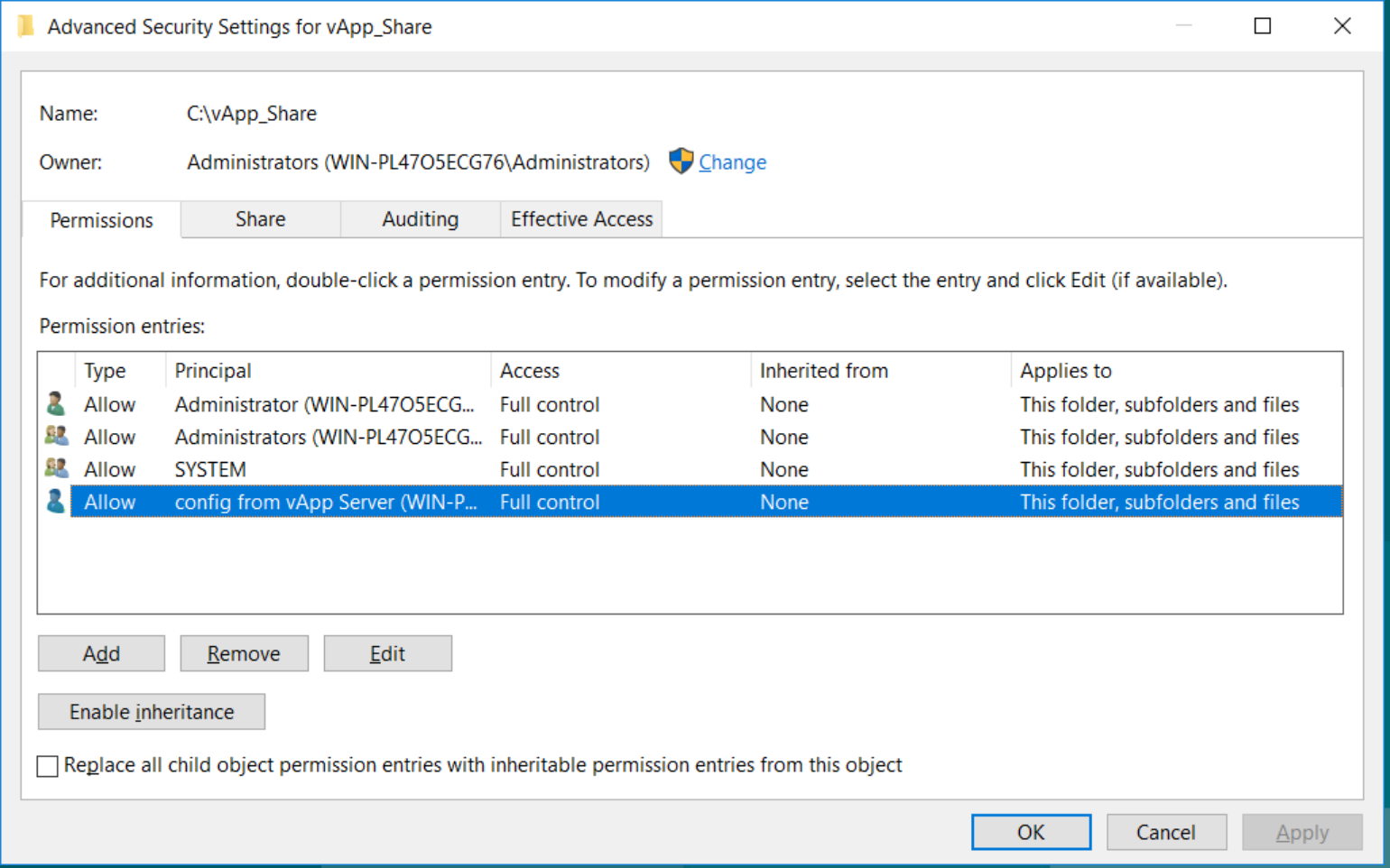




1. If Network discovery and file sharing window pops up, select “No, make the network that I am connected to a private network”



View Permissions for ‘config’ service ID

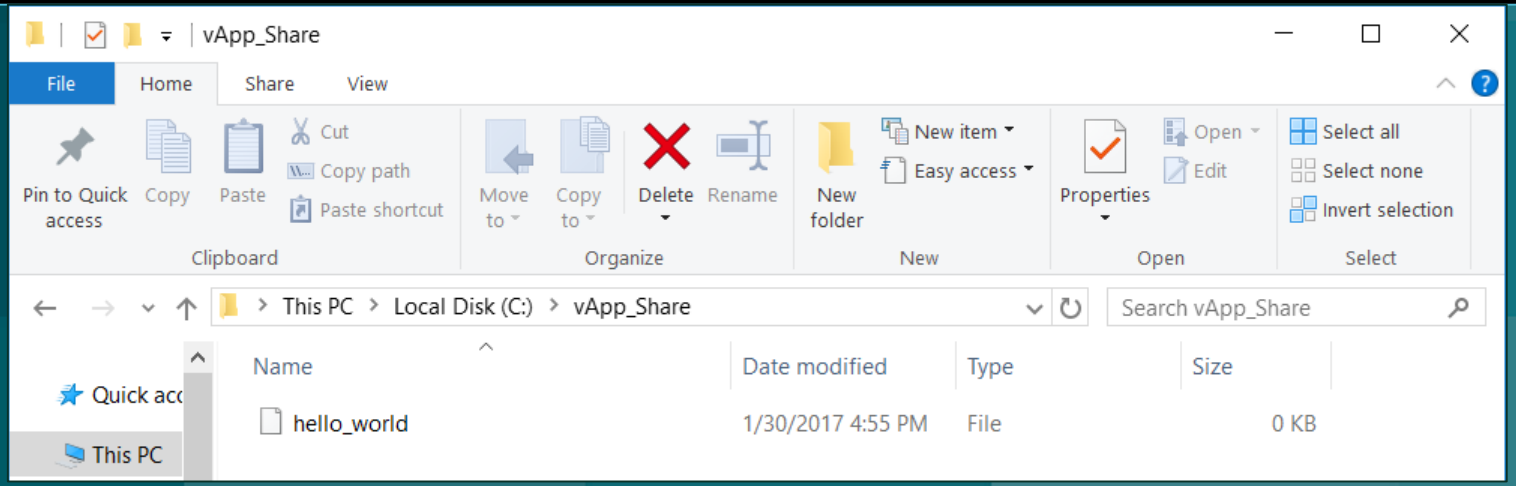


Another view of the service ID's permissions on the network share

Validate service ID from vApp Linux has update access to Network Share

config@vapp001:~/vApp_Share

```
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > pwd
/home/config
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > ls
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > mkdir vApp_Share
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > mount -t cifs -o username=config,password=Password01,uid=500 //192.168.242.136/vApp_Sh
are /home/config/vApp_Share/
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > ls
vApp_Share
config@vapp001 VAPP-14.0.0 (192.168.242.137):~ > cd vApp_Share/
config@vapp001 VAPP-14.0.0 (192.168.242.137):~/vApp_Share > touch hello_world
config@vapp001 VAPP-14.0.0 (192.168.242.137):~/vApp_Share > ls
hello_world
config@vapp001 VAPP-14.0.0 (192.168.242.137):~/vApp_Share >
```



TDM Script Example for Identity Suite vApp

Sections:

1. Initial Variables Defined (hostname, service Id, password,etc.)
2. Cleanup - Remove prior mount points of SAMBA share
3. Create Mount Point of SAMBA share
4. Create PBES format of IM solution management console service ID
5. Update the IM solution's tool: **ImportExportUtility** properties file
6. Execute the IM solution's tool: **ImportExportUtility**
7. Move IM business export files to SAMBA share
8. Update CA Directory DSA config/settings files to allow online DSA backup
9. Re-init CA Directory DSAs to force online DSA backup
10. Export DSA data extract to LDIF format for FULL and TDM (No Passwords)
11. Finish with folder copy, including soft links, of the
    Identity Suite vApp custom folders

1. Initial Variables Defined (hostname, service Id, password,etc.)

```
#!/bin/bash

#### Set Variables for Server
BACKUPDATE=`date +%Y%m%d_%H%M%S_%N`
WIN_SERVER_IP=192.168.242.136
IPADDR=$(ip addr | grep "inet " | egrep "eth0$" | awk '{print $2}' | cut -d "/" -f1)
SHARED_FOLDER=vApp_Share
SHARED_PATH=/home/config/$SHARED_FOLDER/
SHARED_BACKUP_PATH=$SHARED_PATH/vApp_$BACKUPDATE/
SMB_USERID=config
SMB_PASSWORD=Password01
IME_USERID=admin
IME_PASSWORD=CAIMAG1
JAVA_EXE=/opt/CA/jdk1.8.0_71/bin/java
IME_PASSWORD_TOOL=/opt/CA/IdentityManager/IAM_Suite/IdentityManager/tools/PasswordTool
EXPORT_UTIL=/opt/CA/IdentityManager/IAM_Suite/IdentityManager/tools/ImportExportUtility
IME_URL=http:\/\/$IPADDR:8080
IME_DIRECTORIES=UserStore,ProvStore
IME_NAME=identityEnv
```

2. Cleanup - Remove prior mount points of SAMBA share

```
echo "###############################################"
echo "###  01. Remove prior share             ###"
echo "###############################################"
echo "sudo /bin/umount //$WIN_SERVER_IP/$SHARED_FOLDER/     > /dev/null 2>&1"
sudo /bin/umount //$WIN_SERVER_IP/$SHARED_FOLDER/     > /dev/null 2>&1
sudo /bin/umount //$WIN_SERVER_IP/$SHARED_FOLDER/
echo ''
```

3. Create Mount Point of SAMBA share

```
echo "################################################"
echo "### 02.  Create mount point & mount share ###"
echo "################################################"
echo "mkdir $SHARED_PATH   > /dev/null 2>&1"
mkdir $SHARED_PATH        > /dev/null 2>&1
echo "sudo /bin/mount -t cifs -o username=$SMB_USERID,password=$SMB_PASSWORD,uid=500 //$WIN_SERVER_IP/$SHARED_FOLDER  $SHARED_PATH/"
sudo /bin/mount -t cifs -o username=$SMB_USERID,password=$SMB_PASSWORD,uid=500 //$WIN_SERVER_IP/$SHARED_FOLDER  $SHARED_PATH/
echo ''
```

4. Create PBES format of IM solution management console service ID

```
echo "#############################################"
echo "### 04.  Create a PBES Encryption Hash    ###"
echo "#############################################"
echo "# Update Password hash for IME Management User"
cd $IME_PASSWORD_TOOL
PASSWORD_HASH_OUTPUT="$($JAVA_EXE -classpath ../lib/idmutils.jar:../lib/log4j.jar:../lib/cryptojFIPS.jar com.netegrity.rtl.jce.JSafeTools -JSAFE -p
$IME_PASSWORD)"
PASSWORD_HASH=`echo $PASSWORD_HASH_OUTPUT | grep "{PBES}*" | awk '{print $6}'`
echo ""
echo "# The password hash is:  $PASSWORD_HASH "
echo "# for $IME_USERID password =  $IME_PASSWORD"
echo ""
```

5. Update the IM solution's tool: **ImportExportUtility** properties file

```
echo "#########################################################"
echo "### 05. Update IM Import/Export Tool Properties File ###"
echo "#########################################################"
cd $EXPORT_UTIL
# Create fresh backup of orginal config.properties file
cp -r -p config.properties.org config.properties   > /dev/null 2>&1
cp -r -p config.properties config.properties.org    > /dev/null 2>&1
#  Change config file tokens to correct values for IM Export with sed command
#  use single quote for exact match, use double quote to allow string replacements
sed -i "s|baseUrl=http://hostname.mydomain.com:8080|baseUrl=$IME_URL|g"  config.properties
sed -i "s|userName=imuser|userName=$IME_USERID|g" config.properties
sed -i "s|password={PBES}:HUkQTOZbkIs=|password=$PASSWORD_HASH|g" config.properties
sed -i 's|mode=import|mode=export|g'  config.properties
# sed -i 's|resourceType=ALL|resourceType=ALL|g' config.properties
sed -i "s|directories=cadir,prov_dir|directories=$IME_DIRECTORIES|g"  config.properties
sed -i "s|environment=env|environment=$IME_NAME|g"   config.properties
sed -i "s|roleDefFileName=env-RoleDefinitions|roleDefFileName=$IME_NAME-RoleDefinitions|g"  config.properties
# Address double backslash with single quote in sed; then replace with correct token value
sed -i 's|localPath=C:\\\\IME\\\\Temp|localPath=|g'  config.properties
sed -i "s|localPath=|localPath=$SHARED_PATH|g"  config.properties
#sed -i 's|timeout=10|timeout=10|g' config.properties
#sed -i 's|restartEnv=yes|restartEnv=yes|g' config.properties
cp -r -p config.properties config.properties.$BACKUPDATE
```

6. Execute the IM solution’s tool: **ImportExportUtility**

```
echo "#####################################################"
echo "### 06. Export the IME via IM Import/Export Tool ###"
echo "#####################################################"
# Call the IM Export Tool
. ImportExportUtil.sh
echo ''
# Put the config.properties file back to orginal state
cp -r -p config.properties.org config.properties   > /dev/null 2>&1
```

7. Move IM business export files to SAMBA share

```
echo "#########################################################"
echo "### 07. Rename Exported Files with time-date stamp ###"
echo "#########################################################"
echo ''
echo "Rename the output file with date time-stamp"
mkdir $SHARED_BACKUP_PATH  > /dev/null 2>&1
cd $SHARED_PATH
pwd
#cp -r -p UserStore.xml "$SHARED_BACKUP_PATH/UserStore_$BACKUPDATE.xml"
mv -f UserStore.xml "$SHARED_BACKUP_PATH/UserStore_$BACKUPDATE.xml"
#cp -r -p ProvStore.xml "$SHARED_BACKUP_PATH/ProvStore_$BACKUPDATE.xml"
mv -f ProvStore.xml "$SHARED_BACKUP_PATH/ProvStore_$BACKUPDATE.xml"
#cp -r -p identityEnv.zip "$SHARED_BACKUP_PATH/identityEnv_$BACKUPDATE.zip"
mv -f identityEnv.zip "$SHARED_BACKUP_PATH/identityEnv_$BACKUPDATE.zip"
ls -al $SHARED_BACKUP_PATH
echo ''
```

8. Update CA Directory DSA config/settings files to allow online DSA backup

```
echo "############################################################"
echo "### 08.  Update CA Directory DSA to allow online backup ###"
echo "############################################################"
echo " - Configure CA Directory to provide an data dump (zdb file) while DSA are online"
su - dsa -c 'cp -r -p $DXHOME/config/settings/impd.dxc.org $DXHOME/config/settings/impd.dxc'
su - dsa -c 'cp -r -p $DXHOME/config/settings/default.dxc.org $DXHOME/config/settings/default.dxc'  > /dev/null 2>&1
su - dsa -c 'cp -r -p $DXHOME/config/settings/impd.dxc $DXHOME/config/settings/impd.dxc.org'
su - dsa -c 'cp -r -p $DXHOME/config/settings/default.dxc $DXHOME/config/settings/default.dxc.org'  > /dev/null 2>&1
# Edit the DSA settings file to add in one line.  dump dxgrid-db;
su - dsa -c 'echo "dump dxgrid-db;" >> $DXHOME/config/settings/impd.dxc'
su - dsa -c 'chmod 744 $DXHOME/config/settings/default.dxc'
su - dsa -c 'echo "dump dxgrid-db;" >> $DXHOME/config/settings/default.dxc'
echo ""
```

9. Re-init CA Directory DSAs to force online DSA backup

```
echo "#############################################################################################"
echo "### 09. Re-init all DSA to data dump the CA DSAs for IMCD/Userstore (1) & IMPD (4) ###"
echo "#############################################################################################"
echo " - This make take 5-30 seconds to complete "
su - dsa -c 'dxserver init all'    > /dev/null 2>&1
# View for zdb or zd? (in-progress) files
#su - dsa -c 'find  $DXHOME/data/ -name "*.zd*" '
#su - dsa -c 'find  $DXHOME/backup/ -name "*.zd*" '
echo ""
sleep 10
```

10a. Export DSA data extract to LDIF format for FULL and TDM (No Passwords)

```
echo "######################################################################"
echo "### 10. Export DSA backup/offline zdb data files to LDIF file ###"
echo "######################################################################"
echo "10a. Set DSA profile for CONFIG user to ensure DXHOME variable is used"
echo " - Export will happen after the backup/offline zdb files are fully created"
echo " - This make take 5-60 seconds  to complete "
. /opt/CA/Directory/dxserver/install/.dxprofile
echo ""
###
echo "10b. Set WHILE loop for Main (main) DSA"
until [ -f $DXHOME/data/ca-prov-srv-01-impd-main/ca-prov-srv-01-impd-main.zdb ]
do
    echo " - Waiting till CA Directory has completed online data dump of IMPD main DSA"
    sleep 5
done
sleep 5
echo "10c. Execute dxdumbdb for Main (main) DSA - FULL, TDM-NoPassword"
# Use $DXHOME/backup as intermediate location due to folder permission on vApp Server
su - dsa -c "dxdumpdb -z -f $DXHOME/backup/ca-prov-srv-01-impd-main_FULL_$BACKUPDATE.ldif  ca-prov-srv-01-impd-main"   > /dev/null 2>&1
su - dsa -c "dxdumpdb -z -f $DXHOME/backup/ca-prov-srv-01-impd-main_TDM_No_Passwords_$BACKUPDATE.ldif -x
eTPassword,eTEncryptedPassword,eTExitAuthPassword,eTSelfAdminPassword,eTPreviousPassword,eTPropagatePassword,eTIMPasswordData,eTSyncPasswor
d,eTPropagatePassword,eTPSAgentChangePassword,eTTestPassword     ca-prov-srv-01-impd-main"   > /dev/null 2>&1
sleep 5
echo "10d. Copy LDIF to MS Windows Samba share for Main (main) DSA - FULL, TDM-NoPassword"
cp -r -p $DXHOME/backup/ca-prov-srv-01-impd-main_FULL_$BACKUPDATE.ldif  $SHARED_BACKUP_PATH/
cp -r -p $DXHOME/backup/ca-prov-srv-01-impd-main_TDM_No_Passwords_$BACKUPDATE.ldif  $SHARED_BACKUP_PATH/
```

Repeat for all DSAs

10b. Export DSA data extract to LDIF format for FULL and TDM (No Passwords)

```
echo "10n. Set WHILE loop for Userstore DSA"
until [ -f $DXHOME/backup/UserStore_userstore-01.zdb ]
do
    echo " - Waiting till CA Directory has completed online data dump of IMCD UserStore DSA"
    sleep 5
done
sleep 5
echo "10o. Execute dxdumbdb for Userstore DSA - Full, TDM-NoPassword, TDM-NoPassword_nor_Policies"
su - dsa -c "dxdumpdb -z -f $DXHOME/backup/UserStore_userstore-01_FULL_$BACKUPDATE.ldif   UserStore_userstore-01"   > /dev/null 2>&1
su - dsa -c "dxdumpdb -z -f $DXHOME/backup/UserStore_userstore-01_TDM_No_Passwords_$BACKUPDATE.ldif -x userPassword  UserStore_userstore-01"   >
/dev/null 2>&1
su - dsa -c "dxdumpdb -z -f $DXHOME/backup/UserStore_userstore-01_TDM_No_Pwd_or_Policies_$BACKUPDATE.ldif -x
userPassword,IdentityPolicy,createTimestamp,modifiersName,modifyTimestamp      UserStore_userstore-01"   > /dev/null 2>&1
sleep 5
echo "10p. Copy LDIF to MS Windows Samba share for Userstore DSA - FULL, TDM-NoPassword, TDM-NoPassword_nor_Policies"
cp -r -p $DXHOME/backup/UserStore_userstore-01_FULL_$BACKUPDATE.ldif  $SHARED_BACKUP_PATH/
cp -r -p $DXHOME/backup/UserStore_userstore-01_TDM_No_Passwords_$BACKUPDATE.ldif  $SHARED_BACKUP_PATH/
cp -r -p $DXHOME/backup/UserStore_userstore-01_TDM_No_Pwd_or_Policies_$BACKUPDATE.ldif  $SHARED_BACKUP_PATH/
```

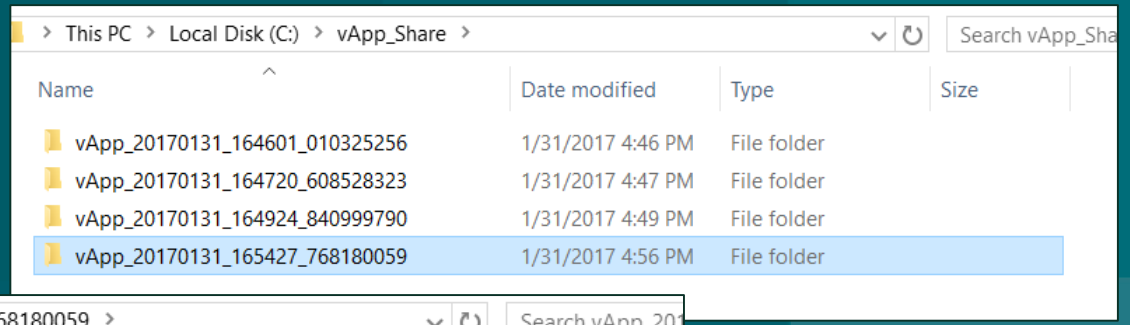
Userstore example

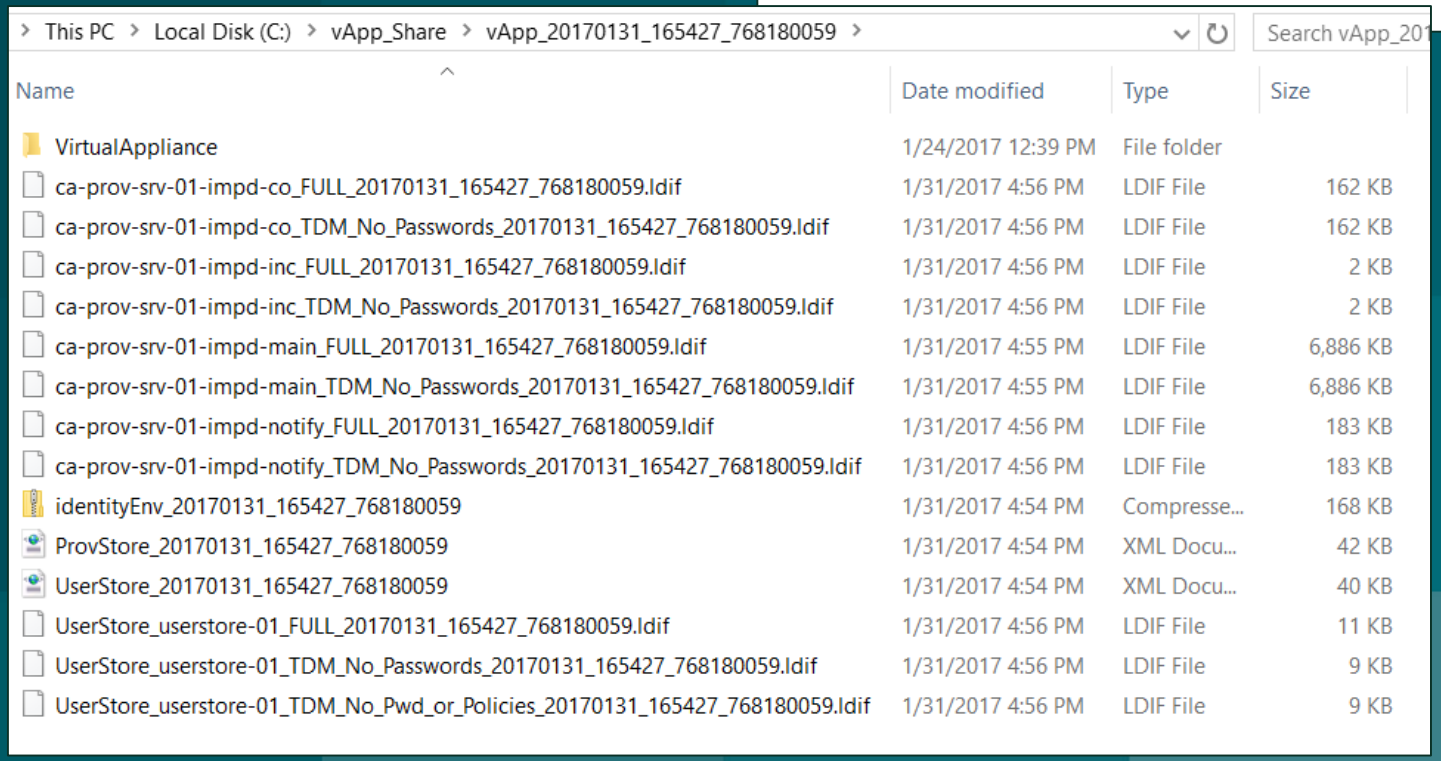
11. Finish with folder copy, including soft links, of the Identity Suite vApp custom folders

```
echo "############################################################"
echo "### 11. Backup Custom Folders for vApp               ###"
echo "############################################################"
echo " -Copy process will follow soft links and return full files"
mkdir $SHARED_BACKUP_PATH   > /dev/null 2>&1
cp -r -p -L /opt/CA/VirtualAppliance $SHARED_BACKUP_PATH     > /dev/null 2>&1

echo ""
echo ""
echo "Size of backup folder: `du -hs  $SHARED_BACKUP_PATH`"
echo ""
echo ""
echo "Done for now"
```

Final State View on MS Windows Server






This is business,
rewritten by software™
ca
technologies

Alan Baugher
Sr. Principal Architect
Alan.Baugher@ca.com

@alanbaugher
636-336-6605
linkedin.com/in/alanbaugher

ca.com