Cloud UAD Solution

BACKGROUND

The UDS Analysis Database, or UAD, is a Microsoft Access application in use at Primary Care Associations (organizations that provide technical assistance to health centers in their state). The UAD application is set up to run on a desktop in the following configuration:

- Folder in use : C:\UAD
- Front end Folder\Name : C:\UAD\UAD.accde
- Backend Folder\Name : C:\UAD\UADdata.accdb

The application runs on a full version of Access (not runtime).

CLOUD UAD

The Cloud UAD version will utilize Windows RDP (Remote Desktop Protocol), and specifically the RemoteApp mode of RDP (more details here : RDP) . RemoteApp is a special mode of RDS, where remote session configuration is integrated into the client operating system and runs a program directly in its own window shell, not displaying anything but the Access app to the local user.

To accomplish the running of the UAD in a RemoteApp setting for the client, there is an architected solution which I will detail here.

SETUP REQUIRED FOR CLOUD UAD

1. Overall Architecture
2. Naming convention
3. Azure VM/ Server setup
4. User Setup on Server
5. Publish to User
6. UAD assumptions

1. Overall Architecture
   - An Azure Virtual Machine (VM) will be created and run to host the UAD database application files (front ends, back end, documents, files created).
   - Each user will run their own Access front end file (UAD.accde file) assigned to their own front end location on the VM.
   - Each client will have a single Access backend database file (UADdata.accdb) that each client user will be linked to on the VM based upon their front end file location.
   - Each user has a unique connection to the server (RDP shortcut), provided to the user, stored on the user's desktop to be run by the user.
   - Each user must have an individual Microsoft 365 login in order to use the other Office features (Excel).
   - Users will run Access in a Runtime environment on the VM.
   - No development will occur on the server (mainly due to Access Runtime limitations).

2. Naming convention
   - Server Admin

    - UADOverlord (me)
- UAD Username
  Each user created will have an Active Directory user created which will be in the following format:
    - <PCA STATE>_PCA<#>
    - PCA STATE = the two character state abbreviation
    - # - the user number
    - Example … LA_PCA1, for Colorado PCA user #1

  Note: the UAD application clients are state-specific organizations, 1 per state, hence the naming convention, albeit limiting, will function for this application. For these examples, I use "LA" for the state Louisiana.

- Front End Location
  For each user, the UAD front end location will be in the following menu path
    - C:\Users\<PCA STATE>_PCA<#>\Documents\UAD.accde
    - Example - C:\Users\LA_PCA1\Documents\UAD.accde

  There can be an unlimited number of users.

- Backend Location
  User data files for the PCA UAD front end file(s) will be directed to the following location:
    - C:\UAD\DB\<PCA STATE>\UADdata.accdb
    - Example - LA data will be stored in C:\UAD\DB\LA\UADdata.accdb

- Report File Location
    - Stored local to the user (current)
    - Example - LA Reports Local to user : C:\Users\LA_PCA1\Documents\SavedReports\
    - Ability to publish reports to state folder (future)
      Example - LA Reports Shared for State : C:\UAD\DB\LA\SavedReports\

3. Azure VM / Server setup
   - Azure VM
     - Set VM to not use Accelerated Networking (reason for setting here)
     - Set IP as static (using IP in connection strings)
   - Open IE to install Chrome or Edge
     - Need to enable file downloads in Internet Explorer
     - Set as default, pin to taskbar
   - Download and install Office
     - Cleanup first in VM – remove Teams Machine-Wide Installer and OneDrive
     - Set up shared computer activation
     - Each user needs to be part of the Remote Desktop Users built in group in order to run Office programs at the same time
     - Each user MUST have at minimum a Microsoft 365 Business Premium account (at present, $20/month). This account allows for shared computer activation. I do not provide this account; I expect my clients to already have this type of account. For other situations, you could provide this for each client but would have to have your clients under one organization login and set this licensing on the server instead.
   - Remote App Tool download and installation
   - Adobe Acrobat Reader DC (UAD reporting), update
     - Install for ALL remote users
   - PDFill Tools download (UAD Report Compilation)
     - Install for ALL remote users
   - Server Manager setup items

    - Join to a Domain – Add Active Directory Domain Services
        - Forrest - UADCloud.local
    - Active Directory
        - Install the Active Directory module for PowerShell
        - Remote Server Administration Tools for Windows 10
    - Install Remote Desktop Services
        - single server deployment
            - single server deployment does not need gateway load balancing the connections because there is only one server
        - Remote Desktop Licensing – purchase licenses, per user license on the VM.
- Other Tools
    - AccessEnum … for displaying NTFS permission structure
    - Windows Admin Center
    - libiconv2.dll (not in Windows Server 2019)
- Access Runtime installation

4. User Setup on Server
    - Set up OU for UADUsers (OU Setup Example)
    - Create UAD users using naming convention, PowerShell (User Setup Examples)
    - Authorize account for remote login - AD DS, User, Member of, RDU
    - Set up permissions (can use PowerShell) ... disable inheritance on certain objects
      Please see the Attachment "User Permissions on VM" for permissions to apply to new users
    - Login to set up user folder
        - I have not been able to programmatically create new Windows user's profile folder (folder with Documents folder, Favorites folder, etc.)
        - To create the user's Windows user profile, log in once as the user
    - Set up trusted locations – group policy
        - Import Office 2016 template into Group Policy
        - Apply trusted locations group policy to UADUser
        - Set user disconnect/logoff group policy (5 minutes for logoff from disconnect)
    - Limit users to application only
        - Normally, if the user wanted to access the remote server via desktop instead of the app, they could change the RDP file as follows
            - Changing remoteapplicationmode:i:1 → 0
            - Removing the following RDP file entries:
                - alternate shell:s:rdpinit.exe
                - remoteapplicationname:s:UDS Analysis Database
                - remoteapplicationprogram:s:||UAD_LA1
        - To thwart this, follow the setup in the following link : restrict a remote desktop user to a single application. Or, see the attachment included here.
        - IT IS IMPORTANT NOT TO LOCK OUT THE ADMIN ACCOUNT FOR THE VM!
          The way to avoid this is make sure that the following setup guidelines are followed:
            - Make sure that the Admin account is not in the OU that includes the remote users who will access the RemoteApp. The GPO will be assigned to the OU.
            - Local domain should show two User categories
                - Users … with Admin to VM, Guest, krbtgt accounts
                - Your OU … includes all the app users. For the UAD cloud, this OU is called UADUsers
            - Admin for the VM can be a member of the Administrators, Domain Admins, Domain Users, Group Policy Creator Owners, etc.

        - Your app users should be attached to the built-in Remote Desktop Users group, and a group for licenses
    - GPO
        - Group policy should be assigned only to the User Configuration for the OU. The name in use for the UAD cloud server is UADApp
        - Follow the path for the OU specified in the link previously mentioned. Here is the path:
          User Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Remote Desktop Session Environment.
        - "Start a program on connection" – Enabled / Program Path and file name will be the program to start when a user attached to this OU logs on.
          NOTE – for the UAD, using the path of the Access Runtime will allow the remote user to open the app, regardless of whether they are trying to open the RemoteApp or trying to open the remote desktop directly.
        - Always show desktop on connection – Disable
        - Make sure that the GPO (UADApp) is enforced.
        - Under the local domain /GPO, you should see the policy. As well, because it is linked to the OU, you should see that the policy is grouped under the OU, too.
    - Outcome
        - With this configuration, the client will not be able to access the desktop view of the VM by using the previously mentioned modifications to the RDP file.
        - The user's experience trying to open the UAD with a modified RDP file will be that they open a blank Access runtime window. Access Runtime, because it is not opened with the UAD application, will not let them navigate any further, giving them an error message. Once they click "OK" to the message, the session will disconnect.
- Obscure settings
    - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management – PoolUsageMaximum and Paged Pool Size

5. Publish to User
    - Create a user-specific RDP connection file using the RemoteApp tool
    - Send new shortcut to client
    - Other instructions to come….

6. UAD Application Assumptions
   The UAD app will run on the Azure VM server. Here are the assumptions when running it:
    - The user runs the runtime version of Access
    - UAD DB storage area is C:\UAD for all users.
        - DBs in C:\UAD\DB\<STATE>
        - Documents in C:\UAD\Documents\
        - Error files (Future) in C:\UAD\ErrorTables\
    - The user runs their own copy of the UAD front end
        - CurrentProject.Path = C:\Users\<STATE>_PCA<#>\Documents\UAD.accde
    - The user connects to a common backend datafile for each PCA
        - PCA Data area - C:\UAD\DB\<STATE>\UADdata.accdb
    - Import files are saved in the common PCA folder, separated by year
        - Import file area - C:\UAD\DB\<STATE>\ImportDataFiles\<year>\
    - Saved UAD reports can be stored locally or in the common PCA folder
        - Local Saved Reports - C:\Users\<STATE>_PCA<#>\Documents\SavedReports
    - Common Saved Reports (future functionality) - C:\UAD\DB\<STATE>\SavedReports\

- User will use their Microsoft 365 login (Microsoft 365 Business Premium, or higher). Token will insert into user folder:
    - C:\Users\<STATE>_PCA<#>\AppData\Local\Microsoft\Office\16.0\Licensing
    - Access will be valid for 30 days, and check when near end of 30 days to confirm. See more details here.

USER SETUP POWERSHELL COMMANDS

Open up Active Directory PowerShell module. Commands to use in this module:

Syntax :

Get-Command New-ADUser –Syntax

Get List of Users (UAD only) :

Get-ADUser -Filter * -Properties samAccountName -SearchBase 'OU=UADUsers,DC=UADUsers,DC=local' | select samAccountName | Sort-Object –Property samAccountName

Get User Info (LA_PCA1):

Get-ADUser LA_PCA1 -Properties CanonicalName, Enabled, GivenName, Surname, Name, UserPrincipalName, samAccountName, whenCreated, PasswordLastSet | Select CanonicalName, Enabled, GivenName, Surname, Name, UserPrincipalName, samAccountName, whenCreated, PasswordLastSet

Example output would be (for UAD username LA_PCA1) :

```
CanonicalName       : UADUsers.local/Users/LA_PCA1
Enabled             : True
GivenName           :
Surname             :
Name                : LA_PCA1
UserPrincipalName   : LA_PCA1@UADUsers.local
samAccountName      : LA_PCA1
whenCreated         : 12/18/2020 1:17:15 PM
PasswordLastSet     : 2/23/2021 3:51:09 PM
```

Add new UAD User (LA_PCA1):

New-ADUser -Name "LA_PCA1" -Path “OU=UADUsers,DC=UADUsers,DC=local” -SamAccountName "LA_PCA1" -UserPrincipalName "LA_PCA1@UADUsers.local" -AccountPassword(Read-Host -AsSecureString "Input Password") -Enabled $true -ChangePasswordAtLogon $false

Disable User (LA_PCA1):

Set-AdUser -Identity LA_PCA1 -Enabled $False

Remove User (LA_PCA1):

This command removes the user with SAM account name " LA_PCA1"

Remove-ADUser -Identity LA_PCA1

Other Run Commands to Help with User Setup:

- secpol. msc
- dsa. msc … Active directory users and computers
- services. msc … windows services
- compmgmt. Msc … computer management

- gpedit. Msc … group policy management
- lusrmgr.msc … not used, domain controllers run user/groups through AD

Who is logged on?

- query user … get session ID
- logoff <session ID>

USER PERMISSIONS ON VM

UAD Storage (C:\UAD\) folder permissions

Administrators – should have Full Control (Read/Write/Special) to all areas

Top Level (C:\UAD\)

- Each user has Full Control (Read/Write/Execute) to the top folder (CHECK THIS)
- Permissions ARE NOT inherited in these areas

Database area (C:\UAD\DB\)

- Each user has Full Control (Read/Write/Execute) to this folder (CHECK THIS)
- Permissions ARE inherited from C:\UAD\

Database areas - PCAs (C:\UAD\DB\<STATE>)

- Each PCA client's user will have Full Control (Read/Write/Execute) to their data folder
    - Example … C:\UAD\DB\LA will allow LA_PCA1, LA_PCA2… etc.
- No other PCAs/users allowed in data areas
- Permissions ARE NOT inherited in these areas

Documents (C:\UAD\Documents\)

- Each user has only READ permissions to the Documents folder

Error files (Future) (C:\UAD\ErrorTables\)

- Each user has only WRITE permissions to the ErrorTables folder

All other areas of the C:UAD folder are not accessible

UAD Front End/App (C:\Users\<STATE>_PCA<#>\Documents) folder permissions

Administrators – should have Full Control (Read/Write/Execute) to all areas

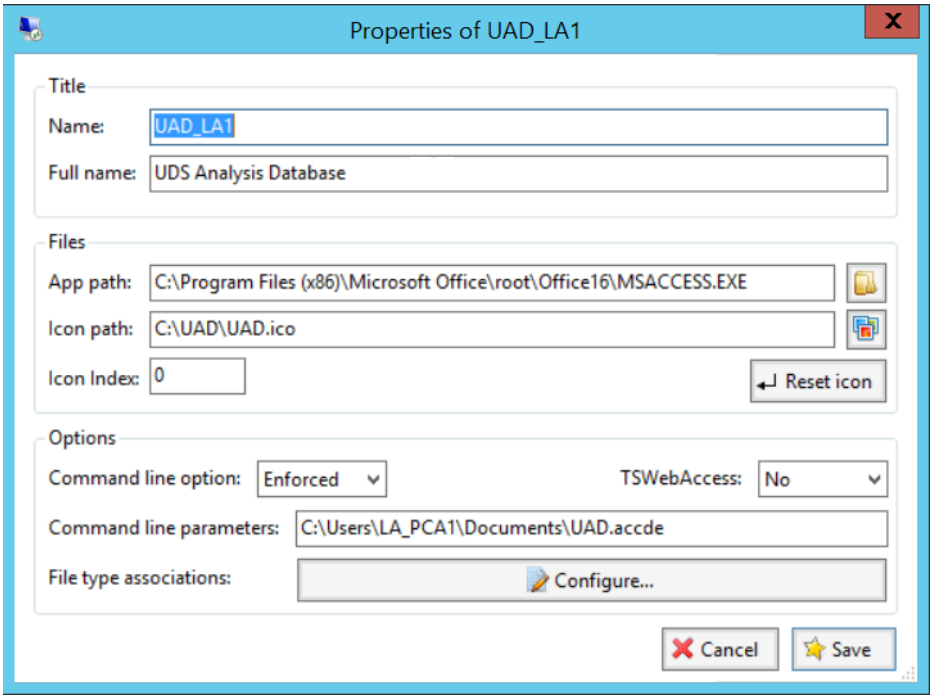
Top Level (C:\Users\)

- UADUSERS has Read/Execute to the top folder (CHECK THIS)
- Permissions ARE NOT inherited in these areas

Front End/App area (C:\Users\< STATE>_PCA<#>\)

- Each user has Full Control (Read/Write/Execute) to their app folder (CHECK THIS)
- Permissions ARE NOT inherited from C:\Users\

REMOTEAPP EXAMPLE



The above screenshot details what should be entered for each new RemoteApp. One of these files needs to be created for each new user (i.e., LA_PCA1, LA_PCA2, …etc.)

Here are the details on the RemoteApp fields:

Name: UAD_<PCA STATE><#>

Full Name: UDS Analysis Database

Path: C:\Program Files (x86)\Microsoft Office\root\Office16\MSACCESS.EXE

Icon Path: C:\UAD\UAD.ico

Icon Index: 0

Command line option: Enforced

TSWebAccess: No

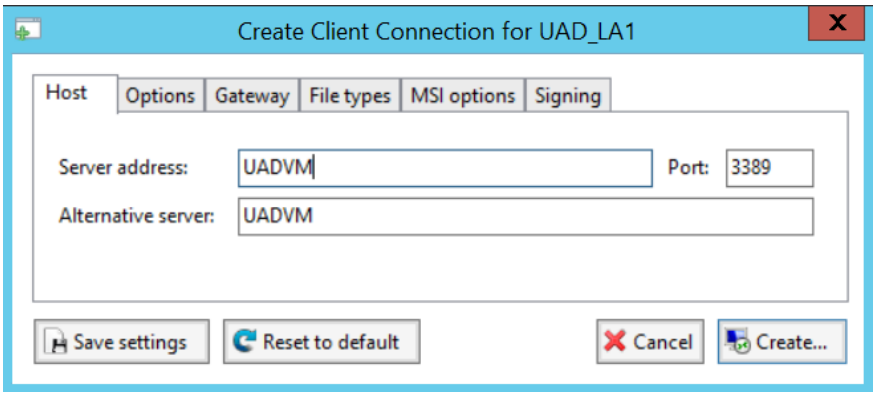
Command line parameters: C:\Users\<PCA STATE>_PCA<#>\Documents\UAD.accde

With the creation of a RemoteApp in the tool, you can then create the RDP connection file. Or, you can just copy an RDP connection file for the client and change a few settings. But the tool is easier.

In the **Command line parameters,** the path will include the UAD Username and point to the front end UAD file set up for each user.

Kim Knight's RemoteApp

RDP CONNECTION FILE EXAMPLE



Once you create the RemoteApp file, you can create the connection file through the RemoteApp tool. This will include a minimal list of items (more comprehensive below), and set the Server/Alternate Server address to a named server. The connection - full address:s:UADVM - does not work (this was set up on the remote computer using the RemoteApp Tool). When I put in the static IP address of the server, this worked. There may be another setup to apply to this in order to apply the IP address via the nickname (UADVM)…

Check "Manually edit RDP file" if you want to edit the file in Notepad after completing this process.

The following is the content on a connection shortcut, created with the RemoteApp tool (edited or revised by opening up in Notepad):

```
allow desktop composition:i:1
allow font smoothing:i:1
alternate full address:s:<IP Address>
alternate shell:s:rdpinit.exe
devicestoredirect:s:*
disableremoteappcapscheck:i:1
drivestoredirect:s:*
full address:s:<IP Address>
username:s:<UAD Username>
prompt for credentials:i:1
prompt for credentials on client:i:1
promptcredentialonce:i:0
redirectcomports:i:1
redirectdrives:i:1
remoteapplicationmode:i:1
remoteapplicationname:s:UDS Analysis Database
remoteapplicationprogram:s:||UAD_LA1
server port:i: :<Port #>
span monitors:i:1
use multimon:i:1
screen mode id:i:2
session bpp:i:32
winposstr:s:0,3,0,0,800,600
compression:i:1
keyboardhook:i:2
```

audiocapturemode:i:0
videoplaybackmode:i:1
connection type:i:7
networkautodetect:i:1
bandwidthautodetect:i:1
displayconnectionbar:i:1
enableworkspacereconnect:i:0
disable wallpaper:i:0
disable full window drag:i:1
disable menu anims:i:1
disable themes:i:0
disable cursor setting:i:0
bitmapcachepersistenable:i:1
audiomode:i:0
redirectprinters:i:1
redirectsmartcards:i:1
redirectclipboard:i:1
redirectposdevices:i:0
autoreconnection enabled:i:1
authentication level:i:2
negotiate security layer:i:1
remoteapplicationicon:s:
shell working directory:s:
gatewayhostname:s:
gatewayusagemethod:i:4
gatewaycredentialssource:i:4
gatewayprofileusagemethod:i:0
gatewaybrokeringtype:i:0
use redirection server name:i:0
rdgiskdcproxy:i:0
kdcproxyname:s:

Notes:

<IP Address> … this will be the static IP of the Azure VM server. Once I am able to know how to set the IP to a domain name, I will apply that instead of the actual static IP

< Port #> … this will be the static inbound port to use for the VM.

<UAD Username> … this does not seem get added with the RemoteApp tool, so may need to add it to the created connection string. This is the username in the format described previously

remoteapplicationname:s:UDS Analysis Database and remoteapplicationprogram:s:||UAD_LA1 … These are the identifiers of the app used with the remote connection. The VM knows where to point the remote connection based upon the setting (in the registry ?) for these entries. Each connection string will be different for each UAD user.

remoteapplicationmode:i:<value> … 1 = RemoteApp, 0 = opens as desktop. In order to avoid having users open the connection directly to the desktop, will need to restrict remote desktop to a single application (detailed here : RDP single app restriction)

username:s:<PCA STATE>_PCA<#> … will need to be added to the file through Notepad

prompt for credentials:i:1 … will need to be added to the file through Notepad

Additionally, anything after "span monitors:i:1" will not get included when creating the RDP connection file through the RemoteApp tool. These lines can be added to the file through Notepad.

Restrict a remote desktop user to a single application on Windows Server

1. Log onto the server as Administrator and run Server Manager.
2. From the Dashboard, select Add roles and features.
3. Add Active Directory Domain Services. Following the prompts.
4. Once Active Directory Domain Services has been installed, you will get a notification that it needs to be configured. Configure it.
5. From the Dashboard, again select Add role and features. This time select Remote Desktop Service installation. Following the prompts.
6. From the Server Manager, select Tools -> Active Directory Domain Users and Computer from the main menu.
7. Add a new Organizational Unit (OU) to the domain server. Next, add your users who will be limited to the one application to the new OU.
8. From the Server Manager, select Tools -> Group Policy Manager from the main menu. Expand your server's domain and select the new OU. Right-click the OU and select Create a GPO in this domain, and link it here... Next, expand the OU and select the new group policy and select Edit. Go to User Configuration -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Remote Desktop Session Environment. Enable and configure Start program on connection. Disable Always show desktop on connection.
9. From the Control Panel, open the System applet. Select Remote Settings. Click Users. click Add. Click Advanced. Select the users added to the new OU and permit them to use Remote Desktop.